

# A Convex Optimization Approach to Discrete Optimal Control

Víctor Valls and Douglas J. Leith  
Trinity College Dublin

**Abstract**—In this paper, we bring the celebrated max-weight features (discrete control actions and myopic scheduling decisions) to the field of convex optimization. In particular, we show that these two features can be encompassed in the subgradient method for the Lagrange dual problem by the use perturbations. One of the appealing features of our approach is that it decouples the choice of a control action from a specific choice of subgradient, and as a result, it allows us to design a range of control policies without changing the underlying convex updates. This allows us to not only consider policies that have a finite (so nonconvex) number of control actions, but also policies that have constraints or costs associated with how actions can be selected.

**Index Terms**—convex optimization, max-weight, subgradient methods, approximate optimisation.

## I. INTRODUCTION

CONVEXITY plays a central role in mathematical optimization from both a theoretical and practical point of view. Some of the significant advantages of formulating a problem as a convex optimization are that there exist numerical methods that can solve the optimization problem in a reliable and efficient manner, and that a solution is always a global optimum. When an optimization problem is not of the convex type, then one enters into the realm of nonconvex optimization where the concrete structure of a problem must be exploited to obtain a solution, often not necessarily the optimal one.

In some special cases there exist algorithms that can find optimal solutions to nonconvex problems. One is the case of max-weight scheduling: an algorithm initially devised for scheduling packets in queueing networks which has received much attention in the networking and control communities in recent years. In short, max-weight was proposed by Tassiulas and Ephremides in their seminal paper [1]. It considers a network of interconnected queues in a slotted time system where packets arrive in each time slot and a network controller has to make a discrete (so nonconvex) scheduling decision as to which packets to serve from each of the queues. Appealing features of max-weight are that the action set matches the actual decision variables (namely, do we transmit or not); that scheduling decisions are made without previous knowledge of the mean packet arrival rate in the system; and that it can stabilize the system (maximize the throughput) whenever there exists a policy or algorithm that would do so. These features have made max-weight popular in the community and have fostered the design of extensions that consider (convex)

utility functions, systems with time-varying channel capacity and connectivity, heavy-tailed traffic, *etc.* Similarly, max-weight has been brought to other areas beyond communication networks including traffic signal control [2], cloud computing [3], economics [4], *etc.*, and has become a key tool for discrete decision making in queueing systems.

However, there are some downsides. The success of max-weight has produced so many variants of the algorithm that the state of the art is becoming not only increasingly sophisticated but also increasingly complex. Furthermore, it is often not clear how to combine the different variants (*e.g.*, utility function minimization with heavy-tailed traffic) since the design of a new control or scheduling policy usually involves employing a new proof or exploiting special structure of the problem. There is a need for abstraction and to put concepts into a unified theoretical framework. While this has been attempted in previous works by means of establishing a connection between max-weight and dual methods in convex optimization, the exact nature of the relationship between these remains unclear, and so the body of work on max-weight approaches is still largely separate from the mainstream literature on convex optimization.

In this paper, we present the first approach that successfully puts the celebrated features of max-weight (discrete control actions and myopic scheduling) firmly within the setting of convex optimization. Our approach consists of formulating the Lagrange dual problem and equipping the subgradient method with a perturbation scheme. To the best of our knowledge, this is the first rigorous work that makes max-weight features available in convex optimization by only using elementary methods and without invoking Foster-Lyapunov techniques or fluid limits. One of the advantages of our approach is that it decouples the choice of subgradient from the selection of control action or decision, which allows us to design control policies with specific properties. For example, policies that select actions from a finite set or that have constraints associated with selecting certain subsets of actions. The finiteness of the action set is of particular significance from a theoretical point of view because we are allowing convex optimization to make nonconvex updates, and from a practical point of view because many systems such as computers make decisions in a discrete-like manner.

The main contributions of the paper are summarized in the following:

- (i) *Unifying Framework*: Our analysis brings the celebrated max-weight features to convex optimization. In particular, they can be encompassed in the subgradient method

for the Lagrange dual problem by using  $\delta_k$  and  $\epsilon_k$  perturbations. The analysis is presented in a general form and provides different types of convergence depending on the statistical properties of the perturbations, including bounds that are not asymptotic.

- (ii) *General Control Policies*: Our analysis clearly separates the selection of a subgradient from a particular choice of control action and establishes the fundamental properties that a control policy should satisfy for it to be optimal.
- (iii) *Discrete Control Policies*: We develop two classes of control policies that allow us to use action sets with a finite number of actions (*i.e.*, so action sets are not convex). The first class of control policy is based on the construction of blocks of actions which allows reordering/shuffling of control actions. The second class is a myopic control policy that can select control actions by looking only at the current system state.

The rest of the paper is organized as follows. We start by introducing the preliminaries, which cover the notation and some background material. In Section III, we study the convergence of the dual subgradient method under a  $(\delta_k, \epsilon_k)$  perturbation scheme, and in Section IV how the  $\epsilon_k$  perturbations can be used to equip the dual subgradient method with discrete actions. Section V provides some remarks and discussion, and Section VI illustrates the power of the results with an example that considers discrete scheduling decisions with constrained actions. Finally, in Section VII we provide a brief overview of the state of the art. All the proofs are in the appendices.

## II. PRELIMINARIES

We start by introducing the notation, some standard convex optimization definitions, and the subgradient method for the Lagrange dual problem.

### A. Notation

The sets of natural, integers and real numbers are denoted by  $\mathbf{N}$ ,  $\mathbf{Z}$  and  $\mathbf{R}$ . We use  $\mathbf{R}_+$  and  $\mathbf{R}^n$  to denote the set of nonnegative real numbers and  $n$ -dimensional real vectors. Similarly, we use  $\mathbf{R}^{m \times n}$  to denote the set of  $m \times n$  real matrices. Vectors and matrices are usually written, respectively, in lower and upper case, and all vectors are in column form. The transpose of a vector  $x \in \mathbf{R}^n$  is indicated with  $x^T$ , and we use  $\mathbf{1}$  to indicate the all ones vector. The Euclidean,  $\ell_1$  and  $\ell_\infty$  norms of a vector  $x \in \mathbf{R}^n$  are indicated, respectively, with  $\|x\|_2$ ,  $\|x\|_1$  and  $\|x\|_\infty$ .

Since we usually work with sequences we will use subscript to indicate an element in a sequence, and parenthesis to indicate an element in a vector. For example, for a sequence  $\{x_k\}$  of vectors from  $\mathbf{R}^n$  we have that  $x_k = [x_k(1), \dots, x_k(n)]^T$  where  $x_k(j)$ ,  $j = 1, \dots, n$  is the  $j$ 'th component of the  $k$ 'th vector in the sequence. For two points  $x, y \in \mathbf{R}^n$  we write  $x \succ y$  when the  $x(j) > y(j)$  for all  $j = 1, \dots, n$ , and  $x \succeq y$  when  $x(j) \geq y(j)$ . We will use operator  $[\cdot]^+$  to denote the projection of a vector  $x \in \mathbf{R}^n$  onto the nonnegative orthant, *i.e.*,  $[x]^+ = [\max\{x(1), 0\}, \dots, \max\{x(n), 0\}]^T$ .

### B. Convex Optimization Problem

Consider the standard constrained convex optimization problem  $\mathcal{P}$

$$\begin{aligned} & \underset{x \in X}{\text{minimize}} && f(x) \\ & \text{subject to} && g_j(x) \leq 0 \quad j = 1, \dots, m \end{aligned} \quad (1)$$

where  $f, g_j : X \rightarrow \mathbf{R}$  are convex functions and  $X$  is a convex subset from  $\mathbf{R}^n$ . We will assume that set  $X_0 := \{x \in X \mid g_j(x) \leq 0, j = 1, \dots, m\} \neq \emptyset$ , and so problem  $\mathcal{P}$  is feasible. Also, and using standard notation, we define  $f^* := \min_{x \in X_0} f(x)$  and  $x^* \in \arg \min_{x \in X_0} f(x)$ .

We can transform problem  $\mathcal{P}$  into an unconstrained convex optimization by applying Lagrange relaxation on the constraints. The Lagrange dual function associated to problem  $\mathcal{P}$  is given by

$$h(\lambda) = \inf_{x \in X} L(x, \lambda) = \inf_{x \in X} \{f(x) + \lambda^T g(x)\},$$

where  $g = [g_1, \dots, g_m]^T$ , and  $\lambda \in \mathbf{R}_+^m$  is a vector of Lagrange multipliers. Function  $h$  is concave [5, Chapter 5] and so we can cast the following unconstrained<sup>1</sup> concave maximisation problem  $\mathcal{D}$

$$\underset{\lambda \succeq 0}{\text{maximize}} \quad h(\lambda)$$

where  $h(\lambda^*) = f^*$  with  $\lambda^* \in \arg \max_{\lambda \succeq 0} h(\lambda)$  when strong duality holds. That is, solving problem  $\mathcal{P}$  is equivalent to solving problem  $\mathcal{D}$ . A sufficient condition for strong duality to hold is the following.

*Assumption 1 (Slater Condition)*:  $X_0$  is non-empty. There exists a vector  $x \in X$  such that  $g_j(x) < 0$  for all  $j = 1, \dots, m$ .

### C. Classic Subgradient Method

Problem  $\mathcal{D}$  can be solved using the subgradient method. One of the motivations for using this iterative method is that it allows the Lagrange dual function to be nondifferentiable and imposes little requirements on the objective function and constraints. Another motivation for using the subgradient method is that when the Lagrangian has favorable structure, then the algorithm can be implemented in a distributed manner, and therefore be used to solve large-scale problems. Nonetheless, in this work, the primary motivation for using the subgradient method in the Lagrange dual problem is that it allows us to handle perturbations on the constraints. As we will show in Section III, this will be key to handle resource allocation problems where the resources that need to be allocated are not known in advance.

*1) Iteration*: In short, the subgradient method for the Lagrange dual problem consists of the following update:

$$\lambda_{k+1} = [\lambda_k + \alpha \partial h(\lambda_k)]^+ \quad k = 1, 2, \dots$$

where  $\lambda_1 \in \mathbf{R}_+^m$ ,  $\partial h(\lambda_k)$  is a subgradient in the subdifferential of  $h$  at point  $\lambda_k$ , and  $\alpha > 0$  is a constant step size. The classic subgradient method can make use of more complex step sizes, but constant step size will play a significant role in our analysis, and it is extensively used in many practical applications.

<sup>1</sup>The Lagrange dual function takes values  $-\infty$  when  $\lambda \prec 0$ .

2) *Computing a Subgradient*: A dual subgradient can be obtained by first minimising  $L(\cdot, \lambda_k)$  and then evaluating  $x_k \in \arg \min_{x \in X} L(x, \lambda_k)$  on the constraints, i.e.,  $\partial h(\lambda_k) = g(x_k)$ . Note that minimising  $L(\cdot, \lambda_k)$  for a fixed  $\lambda_k \in \mathbf{R}_+^m$  is an unconstrained convex optimization that can be carried out with a variety of methods, and the choice of using one method or another will depend on the assumptions made on the objective function and constraints. Sometimes it is not possible to exactly minimize the Lagrangian and an approximation is obtained instead, i.e., an  $x_k \in X$  such that  $L(x_k, \lambda_k) - h(\lambda_k) \leq \xi$  where  $\xi \geq 0$ . This can be equivalently regarded as exactly minimising the Lagrangian when an approximate Lagrange multiplier is used instead of the true Lagrange multiplier (see the Appendix A for a detailed explanation). That is, we obtain an  $x_k \in \arg \min_{x \in X} L(x, \mu_k)$  where  $\mu_k = \lambda_k + \epsilon_k$  where  $\epsilon_k \in \mathbf{R}^m$  such that  $\mu_k \succeq 0$  and can be regarded as a perturbation or “error” in the Lagrange multiplier.

3) *Convergence*: A standard assumption made to prove the convergence of the subgradient method is that the subdifferential of  $h$  is bounded for all  $\lambda \succeq 0$ . We can ensure this by making the following assumption.

*Assumption 2*:  $X$  is bounded.

Observe that since we always have that  $\partial h(\lambda) = g(x)$  for some  $x \in X$ , if  $g(x)$  is bounded for every  $x \in X$  then the subgradients of the Lagrange dual function are also bounded. That is, we have that

$$\|\partial h(\lambda)\|_2 \leq \max_{x \in X} \|g(x)\|_2 := \sigma_g,$$

and  $\sigma_g$  is finite because  $g$  is a closed convex function (and so continuous) and  $X$  is bounded.

The basic idea behind the convergence of the dual subgradient method with constant step size is that

- (i) the Euclidean distance between  $\lambda_k$  and a vector  $\lambda^* \in \Lambda^* := \arg \max_{\lambda \succeq 0} h(\lambda)$  decreases *monotonically* when  $\lambda_k$  is sufficiently “far away” from  $\Lambda^*$ ;<sup>2</sup>
- (ii) when  $\lambda_k$  is sufficiently close to  $\Lambda^*$ , it remains in a ball around it.

Important characteristics of the dual subgradient method are that the size of the ball to which  $\lambda_k$  converges depends on  $\alpha$ ; that  $\lambda_k$  converges to an  $\alpha$ -ball around  $\Lambda^*$  in finite time; and that by selecting  $\alpha$  sufficiently small we can make the  $\alpha$ -ball arbitrarily small. It is important to highlight as well that the monotonic convergence of  $\lambda_k$  to a ball around  $\Lambda^*$  does not imply that the value of the Lagrange dual function improves in each iteration.<sup>3</sup> Yet, since from Assumption 2 we have that the Lagrange dual function is uniformly Lipschitz continuous<sup>4</sup> for  $\lambda \succeq 0$ , if the RHS in

$$|h(\lambda_k) - h(\lambda^*)| \leq \|\lambda_k - \lambda^*\|_2 \sigma_g \quad (2)$$

decreases, then  $h(\lambda_k)$  will eventually approach  $h(\lambda^*)$ .

<sup>2</sup>Under the Slater condition  $\Lambda^*$  is a bounded subset from  $\mathbf{R}_+^m$  (by Lemma 1 in [6]).

<sup>3</sup>By monotonic convergence we mean the Euclidean distance between  $\lambda_k$  and a point in  $\Lambda^*$  decreases.

<sup>4</sup>For any  $\lambda_1, \lambda_2 \succeq 0$  we have that  $|h(\lambda_1) - h(\lambda_2)| \leq \|\lambda_1 - \lambda_2\|_2 \sigma_g$ .

### III. SUBGRADIENT METHOD WITH PERTURBATIONS

In this section, we introduce the framework that will allow us to tackle optimisation problems with discrete control actions. We begin by considering the following convex optimization problem  $\mathcal{P}(\delta)$

$$\begin{aligned} & \underset{x \in X}{\text{minimize}} && f(x) \\ & \text{subject to} && g(x) + \delta \preceq 0 \end{aligned}$$

where  $\delta \in \mathbf{R}^m$ . If perturbation  $\delta$  were known we could use an interior point method or similar to solve the problem. However, we will assume that  $\delta$  is *not known* a priori and therefore the natural way to tackle the problem is to use Lagrange relaxation. The interpretation of perturbation  $\delta$  will depend on the details of the problem being considered, for example, in a packet switched network  $\delta$  may be the (unknown) mean packet arrival rate.

Because the solution of problem  $\mathcal{P}(\delta)$  depends on  $\delta$ , it will be convenient to define  $X_0(\delta) := \{x \in X \mid g(x) + \delta \preceq 0\}$ ,  $f^*(\delta) = \min_{x \in X_0(\delta)} f(x)$ , and  $x^*(\delta)$  to be a solution of problem  $\mathcal{P}(\delta)$ . Similarly, we also parameterise the Lagrangian  $L(x, \lambda, \delta) = f(x) + \lambda^T(g(x) + \delta)$ , the Lagrange dual function  $h(\lambda, \delta) := \inf_{x \in X} L(x, \lambda, \delta)$ , and define the Lagrange dual problem  $\mathcal{D}(\delta)$

$$\underset{\lambda \succeq 0}{\text{maximize}} \quad h(\lambda, \delta)$$

where  $\lambda^*(\delta)$  is a vector in the set of dual optima  $\Lambda^*(\delta) := \arg \max_{\lambda \succeq 0} h(\lambda, \delta)$ .

#### A. Subgradient Method with Perturbations

The general version of the subgradient method we consider is the following

$$\begin{aligned} \lambda_{k+1} &= [\lambda_k + \alpha \partial h(\mu_k, \delta_k)]^+ \\ &= [\lambda_k + \alpha(g(x_k) + \delta_k)]^+ \end{aligned} \quad (3)$$

for  $k = 1, 2, \dots$  with  $\lambda_1 \in \mathbf{R}_+^m$  and where  $x_k \in \arg \min_{x \in X} L(x, \mu_k, \delta_k)$ ,  $\delta_k \in \mathbf{R}^m$  and  $\mu_k = \lambda_k + \epsilon_k$ ,  $\epsilon_k \in \mathbf{R}^m$ . We will refer to  $\delta_k$  and  $\epsilon_k$  as perturbations. Since parameter  $\delta$  is not known in the optimization we have replaced it with a surrogate  $\delta_k$ , which can be regarded as an approximation or perturbed version of parameter  $\delta$ . Later, we will add assumptions on the properties that  $\delta_k$  must have in order that update (3) solves problem  $\mathcal{P}(\delta)$ . An important observation is that for any  $\mu_k \in \mathbf{R}_+^m$

$$\begin{aligned} \arg \min_{x \in X} L(x, \mu_k, \delta) &= \arg \min_{x \in X} \{f(x) + \mu_k^T(g(x) + \delta)\} \\ &= \arg \min_{x \in X} \{f(x) + \mu_k^T g(x)\} \end{aligned}$$

and so  $g(x_k)$  or the partial “subgradient” of  $h(\lambda_k, \delta)$  can be obtained independently of perturbation  $\delta$ .

We are now in position to present the following lemma.

*Lemma 1 (Subgradient Method)*: Consider optimization problem  $\mathcal{P}(\delta)$  and update (3) with  $\mu_k \succeq 0$  for all  $k$ . Suppose  $\{\delta_k\}$  is a sequence of points from  $\mathbf{R}^m$  such that  $\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \delta_i = \delta$ . Then,

$$-\frac{\|\lambda_1 - \theta\|_2^2}{2\alpha k} - \Gamma \leq \frac{1}{k} \sum_{i=1}^k h(\lambda_i, \delta) - h(\theta, \delta) \quad (4)$$

where  $\theta$  is any vector from  $\mathbf{R}_+^m$ ,  $\Gamma := \alpha(\Gamma_a + \Gamma_b + \Gamma_c) + \Gamma_d + \Gamma_e$  and

$$\Gamma_a := \frac{1}{2k} \sum_{i=1}^k \|g(x_i) + \delta\|_2^2, \quad \Gamma_b := \frac{1}{2k} \sum_{i=1}^k \|\delta_i - \delta\|_2^2 \quad (5)$$

$$\Gamma_c := \frac{1}{k} \sum_{i=1}^k (\delta_i - \delta)^T (g(x_i) + \delta) \quad (6)$$

$$\Gamma_d := \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \|g(x_i) + \delta\|_2 \quad (7)$$

$$\Gamma_e := \frac{1}{k} \sum_{i=1}^k (\lambda_i - \theta)^T (\delta_i - \delta) \quad (8)$$

*Proof:* See Appendix B. ■

Lemma 1 is expressed in a general form and establishes a lower bound on  $\frac{1}{k} \sum_{i=1}^k h(\lambda_i, \delta) - h(\theta, \delta)$ , where  $\theta$  is any vector from  $\mathbf{R}_+^m$ . When  $\theta = \lambda^*(\delta)$  we can then upper bound (4) by zero since  $h(\lambda, \delta) \leq h(\lambda^*(\delta), \delta)$  for all  $\lambda \in \mathbf{R}_+^m$ . The bound on the left-hand side of (4) depends on the properties of perturbations  $\delta_k$  and  $\epsilon_k$ . Next, we provide a detailed analysis of the lower bound and how the different assumptions on the properties of the perturbations result in different types of convergence. A summary of the results can be found in Section III-A2.

1) *Analysis of Perturbations:* Firstly, by Assumption 1 and Lemma 1 in [6] we have that  $\lambda^*(\delta)$  is a bounded vector from  $\mathbf{R}_+^m$  and therefore the first term on the LHS of (4)

$$\frac{\|\lambda_1 - \lambda^*(\delta)\|_2^2}{2\alpha k} \quad (9)$$

is bounded and goes to zero as  $k \rightarrow \infty$ . Since (9) is divided by  $\alpha$ , the convergence rate is inversely proportional to the step size used.

Turning now to  $\alpha(\Gamma_a + \Gamma_b + \Gamma_c)$ , when  $\Gamma_a$ ,  $\Gamma_b$  and  $\Gamma_c$  are bounded above then this expression can be made arbitrarily small by selecting  $\alpha$  sufficiently small. Term  $\Gamma_a$  is the sum of the dual subgradients. If we assume that  $X$  is bounded (c.f., Assumption 2) we have that  $\|g(x_i) + \delta\|_2$  is upper bounded by some constant  $\sigma_g$ ,<sup>5</sup> and so  $\Gamma_a$  is bounded by  $\sigma_g^2/2$  for all  $i = 1, \dots, k$ . The bounds on terms  $\Gamma_b$  and  $\Gamma_c$  depend on the characteristics of sequence  $\{\delta_k\}$ . We consider two cases. Case (i)  $\delta_k$ ,  $k = 1, 2, \dots$  are uniformly bounded. Then,  $\Gamma_b$  is trivially uniformly upper bounded for all  $k$ ; and since  $(\delta_i - \delta)^T (g(x_k) + \delta) \leq \|\delta_k - \delta\|_2 \|g(x_k) + \delta\|_2 \leq \|\delta_k - \delta\|_2 \sigma_g$  by Cauchy-Schwarz, we have that  $\Gamma_c$  is also uniformly upper bounded. Case (ii)  $\delta_k$ ,  $k = 1, 2, \dots$  is a realisation of independent random variables with finite variance and kurtosis, but they do not necessarily have to have bounded support. In this case, we can upper bound  $\Gamma_b$  and  $\Gamma_c$  with probability one asymptotically as  $k \rightarrow \infty$  using Hoeffding's inequality [7]. Hoeffding's bound can be applied to  $\Gamma_b$  directly, and for  $\Gamma_c$  it is sufficient to note that  $-\frac{1}{k} \sum_{i=1}^k \sum_{j=1}^m (\delta_i(j) - \delta(j))(g_j(x_k) +$

<sup>5</sup>Note that  $g(x_i) + \delta$  is a subgradient of  $h(\lambda_k, \delta)$ .

TABLE I  
SUMMARY OF THE CONVERGENCE OF THE BOUND IN LEMMA 1  
DEPENDING ON THE ASSUMPTIONS MADE ON THE PERTURBATIONS  
( $\mathbf{E}(\delta_k) = \delta$  FOR ALL  $k$ ,  $\sigma_\delta^2$  IS THE VARIANCE OF  $\delta_k$ , AND  $\sigma_\epsilon^2$  THE  
VARIANCE OF  $\epsilon_k$ ).

	$\sigma_\delta^2 = 0$	$\sigma_\delta^2 < \infty$	$\sigma_\delta^2 = \infty$
$\ \epsilon_k\ _2 < \epsilon$	deterministic	w.h.p.	-
$\sigma_\epsilon^2 < \infty$	w.h.p.	w.h.p.	-
$\sigma_\epsilon^2 = \infty$	$k \rightarrow \infty$	$k \rightarrow \infty$	-

$\delta(j)) \geq -\sum_{j=1}^m \frac{1}{k} \sum_{i=1}^k (\delta_i(j) - \delta(j)) | \sigma_g$  where  $\delta(j)$  is the  $j$ 'th component of vector  $\delta \in \mathbf{R}^m$ .

Finally, we have the terms  $\Gamma_d$  and  $\Gamma_e$  which are not scaled by  $\alpha$  in the bound. Since  $\Gamma_d$  depends on sequence  $\epsilon_k$ , the boundedness of the term will depend on the assumptions we make on this perturbation. We consider three cases. Case (i)  $\|\epsilon_k\|_2 \leq \epsilon$  for all  $k$  for some  $\epsilon > 0$ . In this case we have that  $\Gamma_d$  can be uniformly upper bounded by  $2\epsilon\sigma_g$ . Case (ii)  $\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \|\epsilon_i\|_2 = \epsilon$ . We cannot say anything about  $\Gamma_d$  for finite  $k$ , but we will have that  $\Gamma_d$  is upper bounded by  $2\epsilon\sigma_g$  when  $k \rightarrow \infty$ . An interesting observation is that if  $\{\epsilon_k\}$  were a stochastic process, it would not need to have finite variance in order for  $\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \|\epsilon_i\|_2$  to exist and be finite (note this is in marked contrast to perturbation  $\delta_k$ , which always has to have finite variance). Case (iii)  $\{\|\epsilon_k\|_2\}$  is a realisation of independent random variables with finite variance and mean  $\epsilon$ . In this case we can use Hoeffding's inequality to give a bound on  $\Gamma_d$  with probability one asymptotically as  $k \rightarrow \infty$ .

Term  $\Gamma_e$  is perhaps the term for which the analysis is more delicate. In the deterministic subgradient method we have that  $\delta_k = \delta$  for all  $k$  and so the term is equal to zero for all  $k$ . Observe that when  $\Gamma_e$  is nonnegative, then we can ignore the term since this would still leave a lower bound on the LHS of (4). However, since  $\lambda^*(\delta)$  is not known (we only know it is finite), it is not possible to determine the sign of  $\Gamma_e$ , and so the term could be unbounded below when  $k \rightarrow \infty$ . Because of all this, we will usually require that  $\{\delta_k\}$  is an ergodic process (i.e.,  $\mathbf{E}(\delta_k) = \delta$  for all  $k$ ) and make use of the fact that  $\lambda_k$  and  $\delta_k$  are independent for all  $k$ ; in which case  $\mathbf{E}(\frac{1}{k} \sum_{i=1}^k (\lambda_i - \lambda^*(\delta))^T (\delta_i - \delta)) = \frac{1}{k} \sum_{i=1}^k \mathbf{E}(\lambda_i - \lambda^*(\delta))^T \mathbf{E}(\delta_i - \delta) = 0$  and the *expected* value of the lower bound in Lemma 1 does not depend on term  $\Gamma_e$ .

2) *Summary of the Different Types of Convergence:* Table I provides a summary of the convergence properties of the lower bound obtained in Lemma 1 under the assumption that  $\{\delta_k\}$  is an ergodic stochastic process<sup>6</sup> and expectation is taken with respect to random variable  $\delta_k$  (since we need term  $\Gamma_e$  to vanish). By deterministic convergence we mean that it is possible to obtain a lower bound of  $\frac{1}{k} \sum_{i=1}^k h(\lambda_i, \delta) - h(\lambda^*(\delta), \delta)$  for every  $k = 1, 2, \dots$ ; by w.h.p. that a lower bound can be given with high probability for  $k$  large enough; and by  $k \rightarrow \infty$  that the lower bound will only hold asymptotically, i.e., it is not possible to say anything about the bound for finite  $k$ .

<sup>6</sup>Hence,  $\mathbf{E}(\delta_k) = \delta$  for all  $k$  where  $\delta$  is the perturbation on the constraints given in problem  $\mathcal{P}(\delta)$ .

## B. Recovery of Primal Solutions

We are now in the position to present one of our main theorems, which establishes the convergence of the objective function to a ball around the optimum and the recovery of approximate primal solutions.

*Theorem 1 (Optimality):* Consider problem  $\mathcal{P}(\delta)$  and update

$$\lambda_{k+1} = [\lambda_k + \alpha \partial h(\mu_k, \delta_k)]^+ \quad (10)$$

where  $\mu_k = \lambda_k + \epsilon_k$  with  $\lambda_1 \in \mathbf{R}_+^m$  and  $\{\epsilon_k\}$  a sequence of points from  $\mathbf{R}^m$  such that  $\mu_k \succeq 0$  for all  $k$ . Suppose  $X_0(\delta)$  has nonempty relative interior (the Slater condition is satisfied) and that  $\delta_k$  is an ergodic stochastic process with expected value  $\delta$  and  $\mathbf{E}(\|\delta_k - \delta\|_2^2) = \sigma_\delta^2$  for some finite  $\sigma_\delta^2$ . Further, suppose that  $\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \|\epsilon_i\|_2 = \epsilon$  for some  $\epsilon \geq 0$  and that Assumption 2 holds. Then,

- (i)  $\mathbf{E}(f(\bar{x}_k) - f^*(\delta)) \leq \frac{\alpha M}{2} + \frac{\|\lambda_1\|_2^2}{2\alpha k} + \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g$
- (ii)  $\lim_{k \rightarrow \infty} |\mathbf{E}(f(\bar{x}_k) - f^*(\delta))| \leq \frac{\alpha M}{2} + 2\epsilon \sigma_g$
- (iii)  $\lim_{k \rightarrow \infty} \mathbf{E}(g(\bar{x}_k) + \delta) \leq 0$
- (iv)  $\mathbf{E}\left(\frac{1}{k} \sum_{i=1}^k \lambda_i\right) \prec \infty \quad k = 1, 2, \dots$

where  $\bar{x}_k = \frac{1}{k} \sum_{i=1}^k x_i$  and  $M = \sigma_g^2 + \sigma_\delta^2$ .

*Proof:* See Appendix B. ■

Claim (i) in Theorem 1 establishes that  $\mathbf{E}(f(\bar{x}_k))$  converges to a ball around  $f^*(\delta)$ . The type of convergence will depend on the assumptions made on the perturbations, as indicated in Table I. Also, note that choosing  $\lambda_1 = 0$  is always a good choice to obtain a sharper upper bound. Claim (ii) extends claim (i) by saying that we can obtain a lower bound on the difference  $\mathbf{E}(f(\bar{x}_k) - f^*(\delta))$ , however, in this case, the bound is only asymptotic regardless of the assumptions made on the perturbations. Claim (iii) ensures the primal point recovered from the optimization is asymptotically optimal, and claim (iv) that the expected value of the running average of the Lagrange multipliers is bounded for all  $k$ . Claim (iv) establishes that expected value of the average of the Lagrange multipliers is bounded, and as we will show in Section V it will play an important role in establishing the stability of queueing systems.

Theorem 1 establishes the convergence properties of the dual subgradient update (10) under  $\delta_k$  and  $\epsilon_k$  perturbations. By appropriate definition of these perturbations a wide range of situations can be encompassed. For example, we can use perturbations  $\delta_k$  to relax the perfect knowledge of the constraints, and perturbations  $\epsilon_k$  to capture use of  $\xi$ -subgradients or asynchronism in updates (see [8]). However, of particular interest here is that we can use the  $\epsilon_k$  perturbations in this framework to analyse the use of discrete-valued control actions for solving optimisation problem  $\mathcal{P}(\delta)$ . We consider this in detail in the next section.

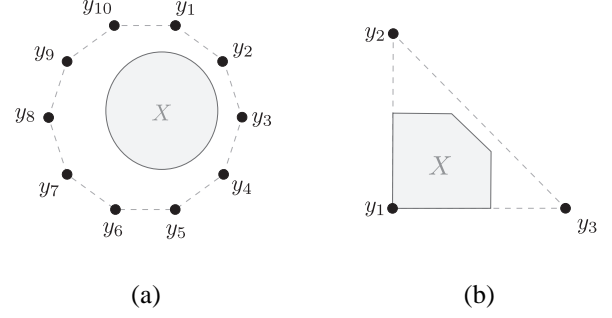


Fig. 1. Illustrating two action sets  $Y$  consisting of a finite collection of points from  $\mathbf{R}^2$ , and two convex sets  $X \subseteq \text{conv}(Y)$ . The convex hulls of  $Y$  are marked in dashed lines.

## IV. DISCRETE CONTROL ACTIONS

In this section, we present the second main contribution of the paper: how to use  $\epsilon_k$  perturbations to equip the dual subgradient method with discrete control actions. Discrete actions or decisions are crucial in control because many systems are restricted to a finite number of states or choices. For instance, a traffic controller has to decide whether a traffic light should be red or green, or a video application which streaming quality to use *e.g.*,  $\{360p, 480p, 720p, 1080p\}$ .

We present two classes of discrete control policies. One that selects discrete control actions in an online or myopic manner based only on the current system state, and another batch approach that chooses discrete control actions in blocks or groups. The latter class is particularly useful for problems where there are constraints or penalties associated with selecting subsets of actions. For example, in video streaming where the application wants to maximize the quality of the video delivered but at the same time minimize the variability of the quality, and so has constraints on how often it can change the quality of the video stream.

### A. Problem Formulation

We start by introducing the following definitions:

*Definition 1 (Finite Action Set):*  $Y$  is a finite collection of points from  $\mathbf{R}^n$ .

*Definition 2 (Convex Action Set):*  $X \subseteq \text{conv}(Y)$  and convex.

We will regard selecting a point  $y$  from finite set  $Y$  as taking a discrete control action. The physical action associated with each point in set  $Y$  will depend on the context in which the optimization is applied, *e.g.*, the points in  $Y$  may correspond to the actions of setting a traffic light to be red and green. Figure 1 shows an example of two sets  $Y$  and respective convex sets  $X \subseteq \text{conv}(Y)$ .

We consider problem  $\mathcal{P}(\delta)$  from Section III, but now require that the inequality constraints are linear, *i.e.*,  $g(x) := Ax$  with  $A \in \mathbf{R}^{m \times n}$ . The reason for this is the following lemma, which is a restatement of [9, Proposition 3.1.2].

*Lemma 2 (Queue Continuity):* Consider updates

$$\lambda_{k+1} = [\lambda_k + \alpha(Ax_k + \delta_k)]^+ \quad (11)$$

$$\mu_{k+1} = [\mu_k + \alpha(Ay_k + \delta_k)]^+ \quad (12)$$

where  $\lambda_1 = \mu_1 \geq 0$ ,  $\alpha > 0$ ,  $A \in \mathbf{R}^{m \times n}$ ,  $\delta_k \in \mathbf{R}^m$ , and  $\{x_k\}$  and  $\{y_k\}$  are, respectively, sequences of points from  $X$  and  $Y$ . Suppose  $\|\sum_{i=1}^k x_i - y_i\|_2 \leq \psi$  for all  $k$  and some  $\psi \geq 0$ . Then,

$$\|\lambda_k - \mu_k\|_2 \leq \epsilon := 2\alpha\|A\|_2\psi \quad k = 1, 2, \dots \quad (13)$$

Lemma 2 says that when the difference in increments  $\|\sum_{i=1}^k x_i - y_i\|_2 \leq \psi$  then  $\mu_k$  is a  $\epsilon$  approximate Lagrange multiplier. Selecting  $x_k$  according to (3) then when  $\|\sum_{i=1}^k x_i - y_i\|_2 \leq \psi$  we can now immediately apply Theorem 1 to conclude that sequence  $\{y_k\}$  of discrete actions (approximately<sup>7</sup>) solves problem  $\mathcal{P}(\delta)$ . This is a key observation. Not only does it (i) establish that we can solve  $\mathcal{P}(\delta)$  using only discrete actions and (ii) give us a testable condition  $\|\sum_{i=1}^k x_i - y_i\|_2 \leq \psi$  that the discrete actions must satisfy, but it also (iii) tells us that *any* sequence  $\{y_k\}$  of discrete actions satisfying this condition solves  $\mathcal{P}(\delta)$ . We therefore have considerable flexibility in how we select actions  $y_k$ . In other words, we have the freedom to select from a range of different optimal control policies without changing the underlying convex updates. One way to use this freedom is to select an optimal control policy that satisfies specified constraints, e.g., that does not switch traffic lights between green and red too frequently or which minimises use of “costly” actions. Such constraints are often practically important yet are difficult to accommodate within classical control design frameworks.

With this in mind, in the rest of this section we now consider methods for constructing sequences  $\{y_k\}$  of discrete actions that stay close to a sequence  $\{x_k\}$  of continuous-valued updates in the sense that  $\|\sum_{i=1}^k x_i - y_i\|_2$  is uniformly bounded. We begin by establishing that for any sequence  $\{x_k \in X \subseteq \text{conv}(Y)\}$  there always exists discrete-valued sequences  $\{y_k \in Y\}$  such that  $\|\sum_{i=1}^k x_i - y_i\|_2$  is uniformly bounded.

### B. Existence of Discrete Sequences

It will prove convenient to exploit the fact that each point  $x \in X$  is the convex combination of points from  $Y$ . Collect the points in  $Y$  as columns in matrix  $W$  and define

$$E := \{v_1, \dots, v_{|Y|}\},$$

$$U := \text{conv}(E) = \{u \in [0, 1]^{|Y|} \mid \mathbf{1}^T u = 1\},$$

where  $v_j$  is an  $|Y|$ -dimensional standard basis vector, i.e., all elements of vector  $v_j$  are equal to 0 except the  $j$ 'th element that is equal to 1, and  $U$  is the  $|Y|$ -dimensional simplex. Since we can always write a vector  $x_i \in X$  as the convex combination of points from  $Y$  there exists at least one vector  $u_i \in U$  such that  $x_i = Wu_i$ .<sup>8</sup> Similarly, there exists a vector  $e_i \in E$  such that  $y_i = We_i$ . Hence,

$$\left\| \sum_{i=1}^k x_i - y_i \right\|_2 = \left\| \sum_{i=1}^k W(u_i - e_i) \right\|_2 \leq \|W\|_2 \left\| \sum_{i=1}^k u_i - e_i \right\|_2$$

<sup>7</sup>Note that provided  $\psi$  is bounded is finite we can make  $\epsilon$  (and so  $\mathbf{E}(f(\bar{x}_k) - f^*(\delta))$  in Theorem 1) arbitrarily small by selecting step size  $\alpha$  sufficiently small.

<sup>8</sup>A vector  $u_i$  can be obtained by minimising  $\min_{u \in U} \|x_i - Wu\|_2^2$ . The non-uniqueness of the solution comes from Carathéodory's theorem—see, for example, [10].

and therefore showing that  $\|\sum_{i=1}^k u_i - e_i\|_2$  is uniformly bounded is sufficient to establish the boundedness of  $\|\sum_{i=1}^k x_i - y_i\|_2$ .

We have the following useful lemma.

**Lemma 3:** Let  $E$  be a set containing the  $|Y|$ -dimensional standard basis vectors,  $U := \text{conv}(E)$ , and  $D := \{\delta \in \mathbf{R}^{|Y|} \mid \delta^T \mathbf{1} = 0, \|\delta\|_\infty \leq 1\}$ . For any vector  $\delta \in D$ , and sequence  $\{u_i\}_{i=1}^{|Y|}$  of points from  $U$ , there exists at least one sequence  $\{e_i\}_{i=1}^{|Y|}$  of points from  $E$  such that

$$(\delta + z - z') \in D, \quad (14)$$

where  $z := \sum_{i=1}^{|Y|} u_i$  and  $z' := \sum_{i=1}^{|Y|} e_i$ . That is,  $\mathbf{1}^T(\delta + z - z') = 0$  and  $\|\delta + z - z'\|_\infty \leq 1$ .

*Proof:* See Appendix C. ■

Let  $\{u_i^{(1)}\}_{i=1}^{|Y|}$  be a sequence of points from  $U$ , and  $\{e_i^{(1)}\}_{i=1}^{|Y|}$  a sequence of points from  $E$  such that  $(z_1 - z'_1) \in D$  where  $z_1 = \sum_{i=1}^{|Y|} u_i^{(1)}$  and  $z'_1 = \sum_{i=1}^{|Y|} e_i^{(1)}$ . By Lemma 3 such a sequence  $\{e_i^{(1)}\}_{i=1}^{|Y|}$  always exists. Similarly, for another sequence  $\{u_i^{(2)}\}_{i=1}^{|Y|}$  of points from  $U$ , we can construct a sequence  $\{e_i^{(2)}\}_{i=1}^{|Y|}$  of points from  $E$  such that  $(z_2 - z'_2 + (z_1 - z'_1)) \in D$  where  $z_2$  and  $z'_2$  are, respectively, the sum of the elements in sequences  $\{u_i^{(2)}\}_{i=1}^{|Y|}$  and  $\{e_i^{(2)}\}_{i=1}^{|Y|}$ . Repeating, it follows that for sequences  $\{u_i^{(\tau)}\}_{i=1}^{|Y|}$ ,  $\tau = 1, 2, \dots, K$  we can construct sequences  $\{e_i^{(\tau)}\}_{i=1}^{|Y|}$ ,  $\tau = 1, 2, \dots, K$  such that

$$\begin{aligned} &(((\dots((z_1 - z'_1) + z_2 - z'_2) + \dots + z_{K-1} - z'_{K-1}) \\ &+ z_K - z'_K)) = \left( \sum_{\tau=1}^K z_\tau - z'_\tau \right) \in D, \end{aligned} \quad (15)$$

where  $z_\tau$  and  $z'_\tau$  are the sum of the elements in the respective sequences. It follows that  $\|\sum_{i=1}^k u_i - e_i\|_2 \leq \sqrt{|Y|}$  for  $k \in \tau|Y|$ ,  $\tau \in \mathbf{Z}_+$  and  $\|\sum_{i=1}^k u_i - e_i\|_2 \leq \sqrt{|Y|(1 + 2|Y|)^2}$  for all  $k = 1, 2, \dots$  (since the sequences can diverge element-wise by at most  $2|Y|$  over the  $|Y|$  steps between  $\tau|Y|$  and  $(\tau + 1)|Y|$ ). We therefore have arrived the following result.

**Theorem 2 (Existence of Discrete Sequences):** For any sequence  $\{u_k\}$  of points from  $U$  there exists a sequence  $\{e_k\}$  of points from  $E$  such that  $\|\sum_{i=1}^k u_i - e_i\|_2$  is uniformly bounded for all  $k = 1, 2, \dots$ .

Note that since we can always permute the entries in sequence  $\{e_k\}$  while keeping  $\|\sum_{i=1}^k u_i - e_i\|_\infty$  bounded, the existence of one sequence implies the existence of many (indeed exponentially many since the number of permutations grows exponentially with the permutation length).

### C. Constructing Sequences of Discrete Actions Using Blocks

We now present our first method for constructing sequences of discrete actions, which uses a block-based approach.

**Lemma 4:** Consider the setup of Lemma 3 and select

$$e_i \in \arg \min_{e \in E} \left\| \left( \delta + z - \sum_{\kappa=1}^{i-1} e_\kappa \right) - e \right\|_\infty, \quad i = 1, \dots, |Y| \quad (16)$$

where  $z := \sum_{i=1}^{|Y|} u_i$  and  $\delta \in D$ . Then,  $-1 \preceq \delta + z - z' \preceq 1$ , with  $z' = \sum_{i=1}^{|Y|} e_i$ .

*Proof:* See Appendix C. ■

Partitioning sequence  $\{u_k\}$ ,  $k = 1, 2, \dots$ , of points from  $U$  into subsequences  $\{u_i^{(\tau)}\}_{i=1}^{|Y|}$ ,  $\tau \in \mathbf{Z}_+$  with  $u_i^{(\tau)} = u_{\tau|Y|+i}$  and applying Lemma 4 recursively yields a sequence  $\{e_i\}$  such that  $\|\sum_{i=1}^k u_i - e_i\|_2$  is uniformly bounded. That is, we have the following.

**Theorem 3:** Let  $\{u_k\}$  be a sequence of points from  $U$  partitioned into subsequences  $\{u_i^{(\tau)}\}_{i=1}^{|Y|}$  with  $u_i^{(\tau)} = u_{\tau|Y|+i}$ ,  $\tau \in \mathbf{Z}_+$ . For  $i = 1, \dots, |Y|$ ,  $\tau = 0, 1, \dots$  select

$$e_i^{(\tau)} \in \arg \min_{e \in E} \left\| \left( z_\tau - \sum_{\kappa=1}^{i-1} e_\kappa^{(\tau)} \right) - e \right\|_\infty \quad (17)$$

where  $z_\tau := \sum_{i=1}^{|Y|} u_i^{(\tau)}$ . Then  $\|\sum_{i=1}^k u_i - e_i\|_2$  is uniformly bounded for all  $k = 1, 2, \dots$ , where  $e_{\tau|Y|+i} = e_i^{(\tau)}$ .

Observe that, with this approach, construction of subsequence  $\{e_i^{(\tau)}\}_{i=1}^{|Y|}$  requires that subsequence  $\{u_i^{(\tau)}\}_{i=1}^{|Y|}$  is known. Hence, we refer to this as a block or batch approach. When sequence  $\{u_k\}$  is observed in an online manner, then sequence  $\{u_k\}$  must be constructed with a delay of  $|Y|$  elements relative to  $\{u_k\}$  since in order to construct  $\{e_i^{(\tau)}\}_{i=1}^{|Y|}$  we must wait for  $|Y|$  elements until  $\{u_i^{(\tau)}\}_{i=1}^{|Y|}$  is observed.

Note that by a similar analysis we can immediately generalise this method of construction to situations where we partition sequence  $\{u_k\}$ ,  $k = 1, 2, \dots$ , into sub-sequences  $\{u_i^{(\tau)}\}_{i=1}^{T_\tau|Y|}$ ,  $T_\tau \in \mathbf{N}$ ,  $\tau \in \mathbf{Z}_+$  with  $u_i^{(\tau)} = u_{\sum_{t=0}^{\tau-1} T_t|Y|+i}$  i.e. where the subsequences can be different lengths so long as they are all some multiple of  $|Y|$ .

**1) Constrained Control Actions:** We can permute sequence  $\{e_i^{(\tau)}\}_{i=1}^{|Y|}$  arbitrarily while keeping  $\|\sum_{i=1}^k u_i - e_i\|_2$  uniformly bounded (since the vectors  $u_i^{(\tau)}$  and  $e_i^{(\tau)}$  both lie in the unit ball then  $\|u_i^{(\tau)} - e_j^{(\tau)}\|_\infty \leq 2$  for all  $i, j \in \{1, \dots, |Y|\}$ ). When the sequence of admissible actions is constrained, this flexibility can be used to select a sequence of actions which is admissible. For example, sequence  $\{0, 1, 0, 1\}$  might be permuted to  $\{0, 0, 1, 1\}$  if the cost of changing the action taken is high.

#### D. Constructing Sequences of Discrete Actions Myopically

We now consider constructing a discrete valued sequence  $\{e_k\}$  in a manner which is myopic or “greedy”, i.e., that selects each  $e_k$ ,  $k = 1, 2, \dots$  by only looking at  $u_i$ ,  $i = 1, \dots, k$  and  $e_i$ ,  $i = 1, \dots, k-1$ . We have the following theorem.

**Theorem 4:** Let  $\{u_k\}$  be a sequence of points from  $U$ . Select

$$e_k \in \arg \min_{e \in E} \|s_{k-1} + u_k - e\|_\infty, \quad (18)$$

where  $s_k = \sum_{i=1}^k (u_i - e_i)$ . Then, we have that  $-1 \preceq s_k \preceq (|Y| - 1)\mathbf{1}$ , and

$$\left\| \sum_{i=1}^k u_i - e_i \right\|_2 \leq C := \sqrt{|Y|}(|Y| - 1) \quad (19)$$

*Proof:* See Appendix C. ■

Theorem 4 guarantees that by using update (18) the difference  $\|\sum_{i=1}^k u_i - e_i\|_2$  is uniformly bounded. Observe that update (18) selects a vector  $e \in E$  that decreases the largest

component of vector  $s_k$ , and so does not provide flexibility to select other actions in  $Y$ . That is, the benefit of myopic selection comes at the cost of reduced freedom in the choice of discrete control action. Regarding complexity, solving (18) in general involves using exhaustive search. However, it is not necessary to solve (18) for every  $u_k$ ,  $k = 1, 2, \dots$  so long as the difference between the steps when (18) is performed is bounded. This is because the elements of  $u_k$  and  $e_k$  can diverge by at most 2 at every step (recall both vectors lie in the unit ball) and so remain bounded between updates. Hence, the cost of (18) can be amortised across steps. We have the following corollary.

**Corollary 1:** Consider the setup of Theorem 4 and suppose update (18) is performed at steps  $k \in \{\tau_1, \tau_2, \dots\} := \mathcal{T} \subseteq \mathbf{N}$ ; otherwise,  $e_k$  is selected equal to  $e_{k-1}$ . Then, we have that  $-\bar{\tau}\mathbf{1} \preceq s_k \preceq \bar{\tau}(|Y| - 1)\mathbf{1}$ , for all  $k$  where  $\bar{\tau} = \max_{j \in \{1, 2, \dots\}} \{\tau_{j+1} - \tau_j\}$  and

$$\left\| \sum_{i=1}^k u_i - e_i \right\|_2 \leq \bar{\tau}C. \quad (20)$$

*Proof:* See Appendix C. ■

**1) Constrained Control Actions:** As already noted, we are interested in constructing sequences of actions in a flexible way that can be easily adapted to the constraints on the admissible actions. Corollary 1 allows us to accommodate one common class of constraints on the actions, namely that once an action has been initiated it must run to completion. For example, suppose we are scheduling packet transmissions where the packets are of variable size and a discrete action represents transmitting a single bit, then Corollary 1 ensures that a sequence of bits can be transmitted until a whole packet has been sent. The condition that  $\bar{\tau}$  must be finite corresponds in this case to requiring that packets have finite length.

In the case of myopic updates it is difficult to give an algorithm without specifying a problem; nevertheless, we can establish the conditions that a generic algorithm should check when selecting a sequence of actions. As shown in the previous section, for finite  $|Y|$  it is possible to construct a sequence of actions that breaks free from the past for a subsequence that is sufficiently large. The same concept can be applied to the myopic case, but now we must ensure that  $\|s_k\|_\infty$  is bounded for all  $k$ . This motivates the following theorem.

**Theorem 5:** Let  $\{u_k\}$  be a sequence of points from  $U$ . For any sequence  $\{e_k\}$  of points from  $E$  we have that

$$\left\| \sum_{i=1}^k u_i - e_i \right\|_2 \leq \gamma_k C \quad (21)$$

where  $\gamma_k = -\min_{j \in \{1, \dots, |Y|\}} s_k(j)$ ,  $s_k = \sum_{i=1}^k u_i - e_i$  and  $C = \sqrt{|Y|}(|Y| - 1)$ .

*Proof:* See Appendix C. ■

Theorem 5 says that when we can construct a sequence of actions  $\{e_k\}$ , such that  $\gamma_k$  is bounded then the difference  $\|\sum_{i=1}^k u_i - e_i\|_2$  will be bounded. However, now  $e_k$  does not need to be obtained as in (18) as long as it is selected with some “care”. Namely, by not selecting actions that decrease lower bound  $\gamma_k$  “excessively”. For example, choosing a vector



$e_k$  that decreases a positive component of vector  $s_k$  will be enough. In addition to providing flexibility to respect constraints in the admissible actions, the implications of this are also important for scalability, namely when action set is large we do not need to do an exhaustive search over all the elements to select a vector from  $E$ .

## V. REMARKS

### A. Discrete Queues

As already noted in the example in Section VI,  $\epsilon_k$  perturbations can be used to model the behavior of discrete valued queues. That is, queues that contain discrete valued quantities such as packets, vehicles or people. Namely, suppose that approximate Lagrange multipliers  $\mu_k$  take a queue-like form, *i.e.*,

$$\mu_{k+1} = [\mu_k + \alpha\gamma_k]^+,$$

where  $\gamma_k \in Y \subset \mathbf{Z}^m$  and represents the number of discrete elements that enter and leave the queue. Next, observe that since  $[\cdot]^+$  is a homogeneous function we can define  $Q_k := \mu_k/\alpha$  and obtain iterate

$$Q_{k+1} = [Q_k + \gamma_k]^+. \quad (22)$$

That is, since  $\gamma_k$  takes values from a subset of integers we have that  $Q_k$  is also integer valued, and therefore update (22) models the dynamics of a queue with discrete quantities. Provided  $\epsilon_k = \lambda_k - \alpha Q_k$  satisfies the conditions of Theorem 1) then we can use  $\alpha Q_k$  (which is equal to  $\mu_k$ ) in update (10).

### B. Queue Stability

A central point in max-weight approaches is to show the stability of the system, which is usually established by the use of Foster-Lyapunov arguments. In our approach it is sufficient to use the boundedness of the Lagrange multipliers (*i.e.*, claim (iv) in Theorem 1) and the fact that difference  $\|\lambda_k - \mu_k\|_2 = \|\lambda_k - \alpha Q_k\|_2$  remains uniformly bounded. Observe that the uniform boundedness of  $\|\lambda_k - \alpha Q_k\|_2$  implies that  $\mathbf{E}(\frac{1}{k} \sum_{i=1}^k \alpha Q_i) < \infty$  for all  $k$  as well. Also, by the linearity of the expectation we can take the expectation inside the summation, and by dividing by  $\alpha$  and taking  $k \rightarrow \infty$  we can write

$$\lim_{k \rightarrow \infty} \frac{1}{k} \sum_{i=1}^k \mathbf{E}(Q_i) < \infty,$$

which is the definition of strong stability given in the literature of max-weight—see, for example, [11, Definition 4].

### C. Primal-Dual Updates

The focus of the paper has been in solving the Lagrange dual problem directly, but the analysis encompasses primal-dual approaches as a special case. For instance, instead of obtaining an  $x_k \in \arg \min_{x \in X} L(x, \lambda_k, \delta)$  in each iteration we could obtain an  $x_k$  that provides sufficient descent *i.e.*, an update that makes the difference  $L(x_k, \lambda_k, \delta) - h(\lambda_k)$  decrease monotonically. This strategy is in spirit very close to the dynamical system approaches presented in [12], [13], [14],

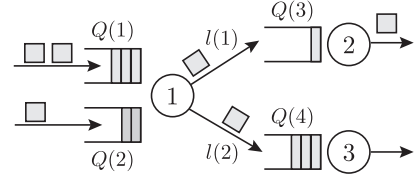


Fig. 2. Illustrating the network in the example of Section VI. The access point (node 1) receives packets in queue 1 and queue 2, and sends them, respectively, to nodes 2 and 3. The packets sent by nodes 2 and 3 leave the system.

which usually require the objective function and constraints to have bounded curvature. In our case, having that the difference  $L(x_k, \lambda_k, \delta) - h(\lambda_k)$  decreases monotonically translates into having a sequence  $\{\epsilon_k\}$  that converges to zero and so the perturbations vanishes.<sup>9</sup>

### D. Unconstrained Optimization

The main motivation for using Lagrange relaxation is to tackle resource allocation problems of the sort tackled by max-weight approaches in the literature. However, our results obtained (Lemma 1) naturally extend to unconstrained optimization problems. In this case  $h$  becomes the unconstrained objective function that takes values from  $\mathbf{R}^m \rightarrow \mathbf{R}$ , the Lagrange multiplier  $\lambda$  is the “primal” variable from a convex set  $D$ , and  $\Pi_D$  (instead of  $[\cdot]^+$ ) the Euclidean projection of a vector  $\lambda \in \mathbf{R}^m$  onto  $D$ . The proof of Lemma 1 remains unchanged, and it is sufficient to note that the Euclidean projection onto a convex set is nonexpansive. That is,  $\|\Pi_D(\lambda_1) - \Pi_D(\lambda_2)\|_2 \leq \|\lambda_1 - \lambda_2\|_2$  for all  $\lambda_1, \lambda_2 \in \mathbf{R}^m$ .

## VI. NUMERICAL EXAMPLE

### A. Problem Setup

Consider the network shown in Figure 2 where an Access Point (AP), labelled as node 1, transmits to two wireless stations, labelled as nodes 2 and 3. Time is slotted and in each slot packets arrive at the queues  $Q(1)$  and  $Q(2)$  of node 1, which are later transmitted to nodes 2 and 3. In particular, node 1 transmits packets from  $Q(1)$  to  $Q(3)$  (node 2) using link  $l(1)$ , and packets from  $Q(2)$  to  $Q(4)$  (node 3) using link  $l(2)$  (see Figure 2). We represent the connection between queues using incidence matrix  $A \in \{-1, 0, 1\}^{n \times l}$  where  $-1$  indicates that a link is leaving a queue;  $1$  that a link is entering a queue; and  $0$  that a queue and a link are not connected. For example, a  $-1$  in the  $j$ 'th element of the  $i$ 'th row of matrix  $A$  indicates that link  $j$  is leaving queue  $i$ . The incidence matrix of the network illustrated in Figure 2 is given by

$$A = \begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}^T. \quad (23)$$

In each time slot, the AP (node 1) takes an action from action set  $Y := \{y(0), y(1), y(2)\} = \{[0, 0]^T, [1, 0]^T, [0, 1]^T\}$ , where each action indicates which link to activate. For simplicity of exposition we will assume that activating a link

<sup>9</sup>See Section II-C2 for a more detailed explanation of how the difference  $L(x_k, \lambda_k, \delta) - h(\lambda_k)$  relates to the  $\epsilon_k$  perturbations.



corresponds to transmitting a packet, and therefore selecting action  $y(1)$  corresponds to transmitting a packet from  $Q(1)$  to node  $Q(3)$ ; action  $y(2)$  to transmitting a packet from  $Q(2)$  to  $Q(4)$ ; and action  $y(0)$  to doing nothing, *i.e.*, not activating any of the links.

The goal is to design a scheduling policy for the AP (select actions from set  $Y$ ) in order to minimize a convex cost function of the average throughput (*e.g.*, this might represent the energy usage), and ensure that the system is stable, *i.e.*, the queues do not overflow and so all traffic is served.

The convex or fluid-like formulation of the problem is

$$\begin{aligned} & \underset{x \in X}{\text{minimize}} && f(x) \\ & \text{subject to} && Ax + b \preceq 0 \end{aligned} \quad (24)$$

where  $f : \mathbf{R}^2 \rightarrow \mathbf{R}$  is a convex utility function,  $X \subseteq \text{conv}(Y)$ ,  $A$  the incidence matrix given in (23), and  $b$  a vector that captures the *mean* exogenous packet arrivals/departures in the system.<sup>10</sup>

### B. Unconstrained Control Actions

Problem (24) can be solved with the optimization framework presented in Section III. That is, with update

$$\lambda_{k+1} = [\lambda_k + \alpha(Ax_k + B_k)]^+ \quad (25)$$

where  $x_k \in \arg \min_{x \in X} \{f(x) + \lambda_k^T Ax\}$ , and note we have replaced  $b$  with random variable  $B_k$  in order to capture the fact that packet arrivals in node 1 might be a realisation of a stochastic process.

Observe from update (25) that by selecting an  $x_k \in X := \text{conv}(Y)$  we obtain the fraction of time each of the links should be used in each iteration, but not which packet to transmit from each of the queues in each time slot. Nonetheless, we can easily incorporate (discrete) control actions  $y_k \in Y$  by using, for example, Theorem 4. Also, note that if we define approximate Lagrange multiplier  $\mu_{k+1} = [\mu_k + \alpha(Ay_k + B_k)]^+$  and let  $Q_k := \mu_k / \alpha$  we obtain queue dynamics

$$Q_{k+1} = [Q_k + Ay_k + B_k]^+,$$

which are identical to those of the real queues in the system. By Theorem 1 we can use update  $x_k \in \arg \min_{x \in X} \{f(x) + \mu_k^T Ax\} = \arg \min_{x \in X} \{f(x) + \alpha Q_k^T Ax\}$ , and with this change we now do not need to compute the Lagrange multipliers  $\lambda_k$  but instead looking at the current queue occupancies  $Q_k$  in the system is enough for selecting control actions.

### C. Constrained Control Actions

We now extend the example to consider the case where the admissible sequence of control actions is constrained. Specifically, suppose it is not possible to select action  $y(1)$  after  $y(2)$  without first selecting  $y(0)$ ; and in the same way, it is not possible to select  $y(2)$  after  $y(1)$  without first selecting

$y(0)$ . However,  $y(1)$  or  $y(2)$  can be selected consecutively. An example of an admissible sequence would be

$$\{y(1), \dots, y(1), y(0), y(2), \dots, y(2), y(0), y(1), \dots, y(1)\}.$$

This type of constraint appears in a variety of application areas and are usually known in the literature as reconfiguration or switchover delays [15], and capture the cost of selecting a “new” control action. In this wireless communication example, the requirement of selecting action  $y(0)$  every time the AP wants to change from action  $y(1)$  to  $y(2)$  and from  $y(2)$  to  $y(1)$  might be regarded as the time required for measuring Channel State Information (CSI) in order to adjust the transmission parameters.<sup>11</sup>

In this case, the constraints on the selection of control actions will affect the definition of set  $X$  in the problem. To see this, observe that if we select sequence of actions in blocks of length  $T|Y|$  with  $T \in \mathbf{N}$ , and  $y(1)$  and  $y(2)$  appear (each) consecutively, then  $y(0)$  should appear *at least* twice in order for the subsequence to be compliant with the transmission protocol. Conversely, any subsequence of length  $T|Y|$  in which  $y(0)$  appears at least twice can be reordered to obtain a subsequence that is compliant with the transmission protocol. For example, if  $T = 3$  and we have subsequence

$$\{y(1), y(2), y(1), y(2), y(2), y(0), y(1), y(2), y(0)\},$$

we can reorder it to obtain

$$\{y(0), y(1), y(1), y(1), y(0), y(2), y(2), y(2), y(2)\},$$

which is a subsequence compliant with the transmission protocol. Since from Section IV-C, we can always choose a subsequence of discrete actions and then reorder its elements, we just need to select a set  $X$  such that  $y(0)$  can be selected twice in a subsequence of  $T|Y|$  elements. This will be the case when every point  $x \in X$  can be written as a convex combination of points from  $Y$  that uses action  $y(0)$  at least fraction  $2/(T|Y|)$ . That is, when

$$X \subseteq \left(1 - \frac{2}{T|Y|}\right) \text{conv}(Y). \quad (26)$$

Observe from (26) that as  $T$  increases we have that  $X \rightarrow \text{conv}(Y)$ , which can be regarded as increasing the capacity of the network since it will be possible to links 1 and 2 a larger fraction of time. Figure 3 illustrates the capacity of the network (set  $X$ ) changes depending on parameter  $T$ .

1) *Simulation:* We run a simulation using  $f(x) = \|Sx\|_2^2$ ,  $S = \text{diag}([1, 3])$ ,  $b = [0.25, 0.5, -1, -1]^T$ ,  $\lambda_1 = \alpha Q_1 = 0$  and  $\alpha = 0.01$ . At each iteration we perform update (25) where  $x_k \in \arg \min_{x \in X} \{f(x) + \alpha Q_k^T Ax\}$ ; and  $B_k(1)$  and  $B_k(2)$  are Bernoulli random variables with mean  $b(1)$  and  $b(2)$  respectively;  $B_k(3)$  and  $B_k(4)$  are equal to  $-1$  for all  $k$  and so the service of nodes 2 and 3 is deterministic. Discrete control actions are selected with update (17) with  $T = 3$ , and so we have that the number of elements in a block or subsequence is 9. The elements in a block are reordered in order to match the protocol constraints on the AP.

<sup>10</sup>More precisely,  $b(1)$  and  $b(2)$  capture, respectively, the mean arrival into  $Q(1)$  and  $Q(2)$ ; and  $b(3)$  and  $b(4)$ , respectively, the mean departure rate from  $Q(3)$  and  $Q(4)$ .

<sup>11</sup>The CSI in wireless communications is in practice measured periodically, and not just at the beginning of a transmission, but we assume this for simplicity. The extension is nevertheless straightforward.

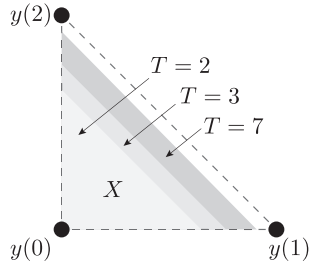


Fig. 3. Illustrating how set  $X$  defined in (26) changes depending on parameter  $T$ . The convex hull of  $Y$  is indicated with dashed line.

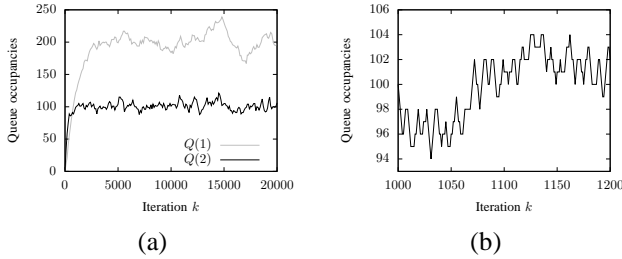


Fig. 4. Illustrating the occupancy of queues 1 and 2 in the network. Figure (b) shows the detail of (a) for an interval of 200 time slots. Observe from (b) that the occupancy of queues is integer valued.

Figure 4 shows the occupancy of the queues in the system, and observe from the detail (b) that the occupancy of the queues is integer valued. Further, observe that  $\alpha Q_k$  converge to a ball around  $\lambda^* = [2, 1]^T$ , the optimal dual variable associated to fluid/convex problem (24). Figure 5 shows the convergence of the objective function for step sizes  $\alpha = \{10^{-2}, 10^{-3}\}$ . Observe from the figure that using a smaller step size heavily affect the convergence time, which is line with the typical accuracy vs. convergence time tradeoff in subgradient methods with constant step size.

## VII. RELATED WORK

Tassiulas and Ephremides introduced the original version of max-weight scheduling in their seminal paper [1], which was later extended by Stolyar [12], Eryilmaz and Srikant [13], and Neely *et al.* [16] to accommodate concave utility functions. When dealing with utility functions, max-weight variants usually start by formulating a fluid or convex optimization, and then use the solution to that problem to design a scheduler for a discrete time system. One of the reasons for considering such an approach is that the fluid or convex formulation

provides the big picture of the problem, whereas max-weight deals with the specific problem at a packet time scale [17, page 9]. Another reason used in the literature is that the dynamics of the Lagrange multipliers (associated with linear inequality constraints) in the dual subgradient method are like queue updates [18, Section 6.1]. Despite previous connections, Huang’s thesis [19] is the first work that establishes rigorous connections between max-weight and convex optimization in the deterministic case. For stochastic problems Huang makes use of the *drift-plus-penalty* algorithm [20], an extension of max-weight scheduling developed by Neely.

Interesting variations of the max-weight for feasibility problems (without utility) include the work in [15], where Çelik *et al.* show that the original max-weight fails to stabilize a system when there are reconfiguration delays associated with selecting new actions/configurations. Their approach to solving the problem is to use a variable frame size max-weight, where actions are allocated in frames to minimize the penalties associated with configuration changes. Another interesting extension is the one proposed by Maguluri *et al.* in [3]. There the authors show that by forcing “inefficiencies” or refresh times it is possible to choose max-weight schedules that maximize the computational capacity of a data server. In [21] Markakis *et al.* show that original max-weight is unstable in the presence of a single heavy-tailed flow, and develop a max-weight scheduling variant that stabilizes a single hop network in the presence of heavy-tailed traffic.

Subgradient methods for solving nondifferentiable convex optimization problems have been studied extensively under various step size rules by Polyak [22], Ermoliev [23] and Shor [24]. Nemirovski and Yudin firstly considered primal averaging in a primal-dual subgradient method in [25], and later by Shor [26], Larson [27] and Nedić [6]. The inexact computation of the subgradient has been treated in previous work in terms of  $\epsilon_k$ -subgradients [28] or deterministic noise [29] in the dual updates.

## VIII. CONCLUSIONS

We have brought the celebrated max-weight features (discrete control actions and myopic scheduling decisions) to the field of convex optimization. In particular, we have shown that these two features can be encompassed in the subgradient method for the Lagrange dual problem by the use of  $\delta_k$  and  $\epsilon_k$  perturbations, and have established different types of convergence depending on the statistical properties of the perturbations. One of the appealing features of our approach is that it decouples the selection of a control action from a particular choice of subgradient, and so allows us to design a range of control policies without changing the underlying convex updates. We have proposed two classes of discrete control policies: one that is able to make discrete actions by looking only at the system’s current state, and another which selects actions using blocks. The latter class is useful for handling systems that have constraints on how actions can be chosen. We have illustrated this with a wireless scheduling example where there are constraints on how packets can be transmitted.

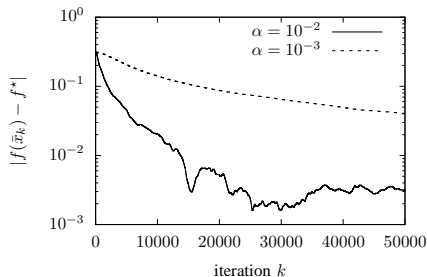


Fig. 5. Illustrating the difference  $|f(\bar{x}_k) - f^*|$  for different step sizes.

## APPENDIX A PRELIMINARIES

### A. Subgradient Convergence

Let  $x_k \in \arg \min_{x \in X} L(x, \lambda_k)$  and observe that we can write (i)  $h(\lambda_k) - h(\lambda^*) = L(x_k, \lambda_k) - L(x^*, \lambda^*) \leq L(x^*, \lambda_k) - L(x^*, \lambda^*) = (\lambda_k - \lambda^*)^T g(x^*) \leq \|\lambda_k - \lambda^*\|_2 \|g(x^*)\|_2$  and (ii)  $h(\lambda_k) - h(\lambda^*) = L(x_k, \lambda_k) - L(x^*, \lambda^*) \geq L(x_k, \lambda_k) - L(x_k, \lambda^*) = (\lambda_k - \lambda^*)^T g(x_k) \geq -\|\lambda_k - \lambda^*\|_2 \|g(x_k)\|_2$ . Further, since  $\|g(x)\|_2 \leq \sigma_g$  for all  $x \in X$  by Assumption 2 we have

$$|h(\lambda_k) - h(\lambda^*)| \leq \|\lambda_k - \lambda^*\|_2 \sigma_g, \quad (27)$$

and so now it is easy to see that if difference  $\|\lambda_k - \lambda^*\|_2$  decreases then the difference  $|h(\lambda_k) - h(\lambda^*)|$  must eventually also decrease.

### B. $\epsilon_k$ -subgradients

In this section, we show how the use of  $\epsilon_k$ -subgradients is equivalent to approximately minimising the Lagrangian. Similar results can be found in Bertsekas [28, pp. 625], but we include them here for completeness.

Let  $x_k \in \arg \min_{x \in X} L(x, \mu_k)$  and  $\mu_k = \lambda_k + \epsilon$  for some  $\epsilon \in \mathbf{R}^m$ , and observe that  $h(\mu_k) = L(x_k, \mu_k) = L(x_k, \lambda_k + \epsilon) = f(x_k) + (\lambda_k + \epsilon)^T g(x_k) \leq f(x_k) + \lambda_k^T g(x_k) + \|\epsilon\|_2 \|g(x_k)\|_2 \leq f(x_k) + \lambda_k^T g(x_k) + \|\epsilon\|_2 \sigma_g$  where the last inequality follows since  $\sigma_g := \max_{x \in X} \|g(x)\|_2$ . Hence,

$$h(\mu_k) - L(x_k, \lambda_k) \leq \|\epsilon\|_2 \sigma_g. \quad (28)$$

We now proceed to show that  $|h(\mu_k) - h(\lambda_k)|$  is bounded. Consider two cases. Case (i)  $h(\mu_k) < h(\lambda_k)$ . From the concavity of  $h$  we have that  $h(\lambda_k) \leq h(\mu_k) + \partial h(\mu_k)^T (\lambda_k - \mu_k) = h(\mu_k) + \partial h(\lambda_k + \epsilon)^T \epsilon = h(\mu_k) + g(x_k)^T \epsilon$ , and therefore

$$0 \leq h(\lambda_k) - h(\lambda_k + \epsilon) \leq \|\epsilon\|_2 \sigma_g.$$

Case (ii)  $h(\lambda_k) > h(\mu_k)$ . Following the same steps than in the first case we obtain  $h(\mu_k) \leq h(\lambda_k, \delta) - g(x_k)^T \epsilon$ , and therefore  $0 \leq h(\mu_k) - h(\lambda_k) \leq \|g(x_k)\|_2 \|\epsilon\|_2 \leq \|\epsilon\|_2 \sigma_g$ . Combining both cases yields  $|h(\lambda_k) - h(\mu_k)| \leq \|\epsilon\|_2 \sigma_g$ , and using (28) we finally obtain

$$0 \leq L(x_k, \lambda_k) - h(\lambda_k) \leq 2\|\epsilon\|_2 \sigma_g := \xi,$$

where the lower bound follows immediately since  $h(\lambda_k) \leq L(x, \lambda_k)$  for all  $x \in X$ . Hence, the error obtained by selecting  $x_k$  to minimize  $L(x, \mu_k)$  is proportional to the difference between  $\lambda_k$  and  $\mu_k$ .

## APPENDIX B PROOFS OF SECTION III

### A. Proof of Lemma 1

For any vector  $\theta \in \mathbf{R}^m$  we have

$$\begin{aligned} \|\lambda_{k+1} - \theta\|_2^2 &= \|[\lambda_k + \alpha(g(x_k) + \delta_k)]^+ - \theta\|_2^2 \\ &\leq \|\lambda_k + \alpha(g(x_k) + \delta_k) - \theta\|_2^2 \\ &= \|\lambda_k - \theta\|_2^2 + \alpha^2 \|g(x_k) + \delta_k\|_2^2 \\ &\quad + 2\alpha(\lambda_k - \theta)^T (g(x_k) + \delta_k) \\ &= \|\lambda_k - \theta\|_2^2 + \alpha^2 \|g(x_k) + \delta_k\|_2^2 + \alpha^2 \|\delta_k - \delta\|_2^2 \\ &\quad + 2\alpha^2 (\delta_k - \delta)^T (g(x_k) + \delta) \\ &\quad + 2\alpha(\lambda_k - \theta)^T (g(x_k) + \delta_k) \end{aligned} \quad (29)$$

where in the last equation we have used the fact that  $\alpha^2 \|g(x_k) + \delta_k\|_2^2 = \alpha^2 \|g(x_k) + \delta_k - \delta + \delta\|_2^2 = \alpha^2 \|g(x_k) + \delta\|_2^2 + \alpha^2 \|\delta_k - \delta\|_2^2 + 2\alpha^2 (\delta_k - \delta)^T (g(x_k) + \delta)$ . Similarly, observe that  $(\lambda_k - \theta)^T (g(x_k) + \delta_k) = (\lambda_k - \theta)^T (g(x_k) + \delta + \delta_k - \delta) = (\lambda_k - \theta)^T (g(x_k) + \delta) + (\lambda_k - \theta)^T (\delta_k - \delta)$  and since

$$\begin{aligned} &(\lambda_k - \theta)^T (g(x_k) + \delta) \\ &= (\lambda_k - \theta)^T (g(x_k) + \delta) + f(x_k) - f(x_k) \\ &= L(x_k, \lambda_k, \delta) - L(x_k, \theta, \delta) \\ &\leq L(x_k, \lambda_k, \delta) - h(\theta, \delta), \end{aligned} \quad (30)$$

we have

$$\begin{aligned} \|\lambda_{k+1} - \theta\|_2^2 &\leq \|\lambda_k - \theta\|_2^2 + \alpha^2 \|g(x_k) + \delta\|_2^2 \\ &\quad + \alpha^2 \|\delta_k - \delta\|_2^2 + 2\alpha^2 (\delta_k - \delta)^T (g(x_k) + \delta) \\ &\quad + 2\alpha(\lambda_k - \theta)^T (\delta_k - \delta) \\ &\quad + 2\alpha(L(x_k, \lambda_k, \delta) - h(\theta, \delta)) \end{aligned} \quad (31)$$

where (30) follows from the fact that  $h(\theta) = \min_{x \in X} L(x, \theta) \leq L(x_k, \theta)$ . Applying the expansion recursively for  $i = 1, \dots, k$  we have

$$\begin{aligned} \|\lambda_{k+1} - \theta\|_2^2 &\leq \|\lambda_1 - \theta\|_2^2 + \alpha^2 \sum_{i=1}^k \|g(x_i) + \delta\|_2^2 \\ &\quad + \alpha^2 \sum_{i=1}^k (\|\delta_i - \delta\|_2^2 + 2(\delta_i - \delta)^T (g(x_i) + \delta)) \\ &\quad + 2\alpha \sum_{i=1}^k (\lambda_i - \theta)^T (\delta_i - \delta) \\ &\quad + 2\alpha \sum_{i=1}^k (L(x_i, \lambda_i, \delta) - h(\theta, \delta)) \end{aligned} \quad (32)$$

Next, observe that since  $L(x_k, \lambda_k, \delta) = L(x_k, \lambda_k, \delta) - L(x_k, \mu_k, \delta) + L(x_k, \mu_k, \delta) \leq |L(x_k, \lambda_k, \delta) - L(x_k, \mu_k, \delta)| + L(x_k, \mu_k, \delta) = h(\mu_k, \delta) + |L(x_k, \lambda_k, \delta) - L(x_k, \mu_k, \delta)| = h(\mu_k, \delta) + |(\lambda_k - \mu_k)^T (g(x_k) + \delta)| = h(\mu_k, \delta) + |\epsilon_k^T (g(x_k) + \delta)| \leq h(\mu_k, \delta) + \|\epsilon_k\|_2 \|g(x_k) + \delta\|_2 = h(\mu_k, \delta) - h(\lambda_k, \delta) + h(\lambda_k, \delta) + \|\epsilon_k\|_2 \|g(x_k) + \delta\|_2 \leq |h(\mu_k, \delta) - h(\lambda_k, \delta)| + h(\lambda_k, \delta) + \|\epsilon_k\|_2 \|g(x_k) + \delta\|_2 \leq h(\lambda_k, \delta) + 2\|\epsilon_k\|_2 \|g(x_k) + \delta\|_2$ , we have that

$$L(x_k, \lambda_k, \delta) - h(\lambda_k, \delta) \leq 2\|\epsilon_k\|_2 \|g(x_k) + \delta\|_2, \quad (33)$$

and therefore

$$\begin{aligned}
\|\lambda_{k+1} - \theta\|_2^2 &\leq \|\lambda_1 - \theta\|_2^2 + \alpha^2 \sum_{i=1}^k \|g(x_i) + \delta\|_2^2 \\
&\quad + \alpha^2 \sum_{i=1}^k (\|\delta_i - \delta\|_2^2 + 2(\delta_i - \delta)^T(g(x_i) + \delta)) \\
&\quad + 2\alpha \sum_{i=1}^k ((\lambda_i - \theta)^T(\delta_i - \delta)) \\
&\quad + 2\alpha \sum_{i=1}^k (2\|\epsilon_i\|_2 \|g(x_i) + \delta\|_2) \\
&\quad + 2\alpha \sum_{i=1}^k (h(\lambda_i, \delta) - h(\theta, \delta)) \tag{34}
\end{aligned}$$

Rearranging terms and dividing by  $2\alpha k$  yields the stated result.

### B. Proof of Theorem 1

Let  $\theta = \lambda^*(\delta)$  in Lemma 1. From (4) and (33) we can write

$$\begin{aligned}
h(\lambda^*(\delta), \delta) &\geq \frac{1}{k} \sum_{i=1}^k h(\lambda_i, \delta) \\
&\geq \frac{1}{k} \sum_{i=1}^k (L(x_i, \lambda_i, \delta) - 2\|\epsilon_i\|_2 \|g(x_i) + \delta\|_2) \\
&= \frac{1}{k} \sum_{i=1}^k (f(x_i) + \lambda_i^T(g(x_i) + \delta) - 2\|\epsilon_i\|_2 \|g(x_i) + \delta\|_2) \\
&\geq f(\bar{x}_k) + \frac{1}{k} \sum_{i=1}^k (\lambda_i^T(g(x_i) + \delta) - 2\|\epsilon_i\|_2 \|g(x_i) + \delta\|_2),
\end{aligned}$$

where the last equation follows from the convexity of  $f$ . Rearranging terms

$$\begin{aligned}
f(\bar{x}_k) - h(\lambda^*(\delta), \delta) &\leq -\frac{1}{k} \sum_{i=1}^k (\lambda_i^T(g(x_i) + \delta) - 2\|\epsilon_i\|_2 \|g(x_i) + \delta\|_2) \tag{35}
\end{aligned}$$

Now, let  $\theta = 0$  in (29) to obtain

$$\begin{aligned}
\|\lambda_{k+1}\|_2^2 &\leq \|\lambda_k\|_2^2 + \alpha^2 \|g(x_k) + \delta\|_2^2 + \alpha^2 \|\delta_k - \delta\|_2^2 \\
&\quad + 2\alpha^2 (\delta_k - \delta)^T(g(x_k) + \delta) + 2\alpha \lambda_k^T(g(x_k) + \delta_k)
\end{aligned}$$

Using the fact that  $\|g(x_k) + \delta\|_2^2 \leq \sigma_g^2$  for all  $k$  and applying the latter expansion recursively

$$\begin{aligned}
\|\lambda_{k+1}\|_2^2 &\leq \|\lambda_1\|_2^2 + \alpha^2 \sigma_g^2 k + \alpha^2 \sum_{i=1}^k \|\delta_i - \delta\|_2^2 \\
&\quad + 2\alpha^2 \sum_{i=1}^k (\delta_i - \delta)^T(g(x_i) + \delta) \\
&\quad + 2\alpha \sum_{i=1}^k \lambda_i^T(g(x_i) + \delta_i) \tag{36}
\end{aligned}$$

Rearranging terms, dropping  $\|\lambda_{k+1}\|_2$  since it is nonnegative, and dividing by  $2\alpha k$  yields

$$\begin{aligned}
-\frac{1}{k} \sum_{i=1}^k \lambda_i^T(g(x_i) + \delta_i) &\leq \frac{\|\lambda_1\|_2^2}{2\alpha k} + \frac{\alpha \sigma_g^2}{2} + \frac{\alpha}{2k} \sum_{i=1}^k \|\delta_i - \delta\|_2^2 \\
&\quad + \frac{\alpha}{k} \sum_{i=1}^k (\delta_i - \delta)^T(g(x_i) + \delta)
\end{aligned}$$

Combining the last bound with (35), and using the fact that  $h(\lambda^*(\delta), \delta) = f^*(\delta)$  (by strong duality, c.f., Assumption 1) yields

$$\begin{aligned}
f(\bar{x}_k) - f^*(\delta) &\leq \frac{\|\lambda_1\|_2^2}{2\alpha k} + \frac{\alpha \sigma_g^2}{2} + \frac{\alpha}{2k} \sum_{i=1}^k \|\delta_i - \delta\|_2^2 \\
&\quad + \frac{\alpha}{k} \sum_{i=1}^k (\delta_i - \delta)^T(g(x_i) + \delta) \\
&\quad + \frac{1}{k} \sum_{i=1}^k 2\|\epsilon_i\|_2 \|g(x_i) + \delta\|_2
\end{aligned}$$

Taking expectations with respect to  $\delta_i$ ,  $i = 1, 2, \dots, k$  we have  $\mathbf{E}(\|\delta_i - \delta\|_2^2) = \sigma_\delta^2$ , and  $\mathbf{E}((\delta_i - \delta)^T(g(x_i) + \delta)) = 0$  since  $x_i$  and  $\delta_i$  are independent. Therefore,

$$\mathbf{E}(f(\bar{x}_k) - f^*(\delta)) \leq \frac{\alpha(\sigma_g^2 + \sigma_\delta^2)}{2} + \frac{\|\lambda_1\|_2^2}{2\alpha k} + \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g. \tag{37}$$

and we have obtained claim (i).

We now proceed to lower bound (37). Taking expectations with respect to  $\delta_i$ ,  $i = 1, 2, \dots, k$  in Lemma 1, and using the fact that  $\lambda_i$  and  $\delta_i$  are independent so  $\mathbf{E}((\lambda_i - \theta)^T(\delta_i - \delta)) = 0$  and  $\mathbf{E}((\delta_i - \delta)^T(g(x_i) + \delta)) = 0$ , we have

$$\begin{aligned}
&-\frac{\|\lambda_1 - \theta\|_2^2}{2\alpha k} - \frac{\alpha(\sigma_g^2 + \sigma_\delta^2)}{2} - \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g \\
&\leq \mathbf{E} \left( \frac{1}{k} \sum_{i=1}^k h(\lambda_i, \delta) - h(\theta, \delta) \right). \tag{38}
\end{aligned}$$

Next, by the convexity of  $-h(\cdot, \delta)$  we can write

$$\frac{1}{k} \sum_{i=1}^k \mathbf{E}(h(\lambda_i, \delta)) = \mathbf{E} \left( \frac{1}{k} \sum_{i=1}^k h(\lambda_i, \delta) \right) \leq \mathbf{E}(h(\bar{\lambda}_k), \delta)$$

and letting  $\theta = \lambda^*(\delta)$

$$\begin{aligned}
&-\frac{\|\lambda_1 - \lambda^*(\delta)\|_2^2}{2\alpha k} - \frac{\alpha(\sigma_g^2 + \sigma_\delta^2)}{2} - \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g \\
&\leq \mathbf{E}(h(\bar{\lambda}_k, \delta) - h(\lambda^*(\delta), \delta)) \leq 0, \tag{39}
\end{aligned}$$

where the upper bound follows from the fact that  $h(\lambda^*(\delta), \delta) = \sup_{\lambda \succeq 0} h(\lambda, \delta)$ . Next, from the saddle point property of the Lagrangian

$$\begin{aligned}
\mathbf{E}(h(\bar{\lambda}_k), \delta) &\stackrel{(a)}{\leq} \mathbf{E}(L(\mathbf{E}(\bar{x}_k), \bar{\lambda}_k, \delta)) \\
&= \mathbf{E}(f(\mathbf{E}(\bar{x}_k)) + \mathbf{E}(\bar{\lambda}_k)^T(g(\mathbf{E}(\bar{x}_k)) + \delta)) \\
&\stackrel{(b)}{\leq} \mathbf{E}(f(\bar{x}_k)) + \mathbf{E}(\bar{\lambda}_k)^T \mathbf{E}(g(\bar{x}_k) + \delta),
\end{aligned}$$

where the expectation on  $\bar{x}_k$  in the RHS of (a) is taken with respect to  $\delta_i, i = 1, \dots, k$ ; and (b) follows from the convexity of  $f$  and  $g$ . Therefore,

$$\begin{aligned} & -\frac{\|\lambda_1 - \lambda^*(\delta)\|_2^2}{2\alpha k} - \frac{\alpha(\sigma_g^2 + \sigma_\delta^2)}{2} - \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g \\ & - \mathbf{E}(\bar{\lambda}_k)^T \mathbf{E}(g(\bar{x}_k) + \delta) \leq \mathbf{E}(f(\bar{x}_k) - f^*(\delta)). \end{aligned} \quad (40)$$

We need to show that  $\mathbf{E}(\bar{\lambda}_k)^T \mathbf{E}(g(\bar{x}_k) + \delta)$  is upper bounded. Observe first that for any sequence  $\{x_k\}$  from  $X$  we can write

$$\lambda_{k+1} = [\lambda_k + \alpha(g(x_k) + \delta_k)]^+ \succeq \lambda_k + \alpha(g(x_k) + \delta_k),$$

and applying the latter recursively we have that

$$\lambda_{k+1} \succeq \lambda_1 + \alpha \sum_{i=1}^k (g(x_i) + \delta_i).$$

Dropping  $\lambda_1$  since it is nonnegative, dividing by  $\alpha k$ , and using the convexity of  $g$  it follows that

$$g(\bar{x}_k) + \frac{1}{k} \sum_{i=1}^k \delta_i \preceq \frac{\lambda_{k+1}}{\alpha k},$$

and taking expectations with respect to  $\delta_i, i = 1, \dots, k$

$$\mathbf{E}(g(\bar{x}_k) + \delta) \preceq \frac{\mathbf{E}(\lambda_{k+1})}{\alpha k}. \quad (41)$$

Multiplying both sides by  $\mathbf{E}(\bar{\lambda}_k)$  (where the expectation is with respect to  $\delta_i, i = 1, \dots, k$ ) and using Cauchy-Schwarz

$$\begin{aligned} \mathbf{E}(\bar{\lambda}_k)^T \mathbf{E}(g(\bar{x}_k) + \delta) & \leq \frac{\mathbf{E}(\bar{\lambda}_k)^T \mathbf{E}(\lambda_{k+1})}{\alpha k} \\ & \leq \frac{\|\mathbf{E}(\bar{\lambda}_k)\|_2 \|\mathbf{E}(\lambda_{k+1})\|_2}{\alpha k}. \end{aligned}$$

We proceed to show that  $\|\mathbf{E}(\bar{\lambda}_k)\|_2$  is bounded. Since  $\mathbf{E}(h(\bar{\lambda}_k, \delta)) \leq h(\mathbf{E}(\bar{\lambda}_k), \delta)$  from (39) we can write

$$\begin{aligned} & -\frac{\|\lambda_1 - \lambda^*(\delta)\|_2^2}{2\alpha k} - \frac{\alpha(\sigma_g^2 + \sigma_\delta^2)}{2} - \frac{2}{k} \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g \\ & \leq h(\mathbf{E}(\bar{\lambda}_k), \delta) - h(\lambda^*(\delta), \delta) \leq 0. \end{aligned} \quad (42)$$

Therefore, from the uniform Lipschitz continuity of the dual function we have that  $\mathbf{E}(\bar{\lambda}_k)$  is attracted to a ball around  $\lambda^*(\delta)$  and therefore it is bounded. We continue by giving a bound on  $\|\mathbf{E}(\lambda_{k+1})\|_2$ . Taking expectations in (34) with respect to  $\delta_i, i = 1, \dots, k$ , letting  $\theta = \lambda^*(\delta)$ ,  $\|g(x_k) + \delta\|_2 \leq \sigma_g$ , and using the fact that  $\lambda_k$  and  $\delta_k$  are independent for all  $k$ , we have

$$\begin{aligned} & \mathbf{E}(\|\lambda_{k+1} - \lambda^*(\delta)\|_2^2) \\ & \leq \|\lambda_1 - \lambda^*(\delta)\|_2^2 + \alpha^2(\sigma_g^2 + \sigma_\delta^2)k + 2\alpha \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g \\ & \quad + 2\alpha \sum_{i=1}^k (h(\lambda_i, \delta) - h(\lambda^*(\delta), \delta)) \end{aligned}$$

Next, observe that since  $h(\lambda_i, \delta) - h(\lambda^*(\delta), \delta) \leq 0$  for all  $i = 1, \dots, k$  we can write  $\mathbf{E}(\|\lambda_{k+1} - \lambda^*(\delta)\|_2^2) \leq \|\lambda_1 - \lambda^*(\delta)\|_2^2 + \alpha^2(\sigma_g^2 + \sigma_\delta^2)k + 2\alpha \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g$  and by using the convexity of  $\|\cdot\|_2^2$   $\|\mathbf{E}(\lambda_{k+1}) - \lambda^*(\delta)\|_2^2 \leq \|\lambda_1 - \lambda^*(\delta)\|_2^2 + \alpha^2(\sigma_g^2 +$

$\sigma_\delta^2)k + 2\alpha \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g$ . That is,  $\mathbf{E}(\lambda_{k+1})$  is within a ball around  $\lambda^*(\delta)$ . Next, since  $\|\lambda^*(\delta)\|_2^2$  is bounded we can write  $\|\mathbf{E}(\lambda_{k+1})\|_2^2 \leq \|\lambda^*(\delta)\|_2^2 + \|\lambda_1 - \lambda^*(\delta)\|_2^2 + \alpha^2(\sigma_g^2 + \sigma_\delta^2)k + 2\alpha \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g$  and by taking square roots in both sides and using the concavity of  $x^a$  for  $x \in \mathbf{R}_{++}$  with  $0 < a < 1$  (see [5, pp. 71]) we have

$$\begin{aligned} \|\mathbf{E}(\lambda_{k+1})\|_2 & \leq \|\lambda^*(\delta)\|_2 + \|\lambda_1 - \lambda^*(\delta)\|_2 \\ & \quad + \alpha\sqrt{k}\sqrt{\sigma_g^2 + \sigma_\delta^2} + \sqrt{2\alpha \sum_{i=1}^k \|\epsilon_i\|_2 \sigma_g} \end{aligned} \quad (43)$$

where the last inequality follows from the concavity of the square root and the fact that all the terms are nonnegative. Hence,

$$\begin{aligned} & \mathbf{E}(\bar{\lambda}_k)^T \mathbf{E}(g(\bar{x}_k) + \delta) \\ & \leq \|\mathbf{E}(\bar{\lambda}_k)\|_2 \left( \frac{\|\lambda^*(\delta)\|_2 + \|\lambda_1 - \lambda^*(\delta)\|_2}{\alpha k} \right. \\ & \quad \left. + \frac{\sqrt{\sigma_g^2 + \sigma_\delta^2}}{\sqrt{k}} + \frac{\sqrt{2\sigma_g} \sqrt{\frac{1}{k} \sum_{i=1}^k \|\epsilon_i\|_2}}{\sqrt{\alpha}\sqrt{k}} \right) \end{aligned}$$

and so we can use (40) to lower bound (37). Taking limits we obtain

$$\lim_{k \rightarrow \infty} |\mathbf{E}(f(\bar{x}_k) - f^*(\delta))| \leq \frac{\alpha}{2}(\sigma_g^2 + \sigma_\delta^2) + 2\epsilon\sigma_g$$

as claimed in (ii).

Claim (iii) follows from (41) and (43). Claim (iv) follows from (42) and the Lipschitz continuity of the dual function.

## APPENDIX C PROOF OF SECTION IV

### A. Proof of Lemma 3

To start, let  $V = |Y|$  and note we always have  $\mathbf{1}^T(z + \delta) = V$  since  $z$  is the sum of  $V$  elements from  $U$ ,  $u^T \mathbf{1} = 1$  and  $\delta^T \mathbf{1} = 0$  for all  $u \in U$ ,  $\delta \in D$ . Further,  $(z + \delta) \succeq -\mathbf{1}$  since  $\delta \succeq -\mathbf{1}$  and  $z \succeq 0$ . Now, let  $r := (z + \delta)$  and define

$$a = -[-r]^+, \quad b = \lfloor r - a \rfloor, \quad c = r - a - b,$$

where the floor in  $b$  is taken element-wise. That is,  $a \in [-1, 0]^V$ ,  $b \in \{0, 1, \dots, V\}^V$ ,  $c \in [0, 1]^V$ . For example, if  $r = [2.2, -0.2]^T$  then  $a = [0, -0.2]^T$ ,  $b = [2, 0]^T$  and  $c = [0.2, 0]^T$ . Observe,

$$\mathbf{1}^T r = \mathbf{1}^T(a + b + c) = V,$$

and since  $b$  is integer valued  $\mathbf{1}^T b \in \mathbf{Z}_+$ , which implies  $\mathbf{1}^T(a + c) \in \mathbf{Z}_+$ . Next, let  $\mathbf{1}^T b = V - \mathbf{1}^T(a + c) := V'$ , and observe  $b$  can be written as the sum of  $V'$  elements from  $E$ , i.e.,

$$b = \sum_{i=1}^{V'} e_i.$$

Next, since  $-\mathbf{1} \preceq a + c \prec \mathbf{1}$  and  $\mathbf{1}^T(a + c) = V - V' := V''$ , there must exist *at least*  $V''$  elements in vector  $(a + c)$  that are nonnegative. If we select  $V''$  elements from  $E$  that match the

nonnegative components of vector  $(a + c)$  we can construct a subsequence  $\{e_i\}_{i=1}^{V''}$  such that

$$-1 \preceq (a + c) - \sum_{i=1}^{V''} e_i \prec 1.$$

Finally, letting  $z' = \sum_{i=1}^{V'} e_i + \sum_{i=1}^{V''} e_i$  yields the result.

### B. Proof of Lemma 4

First of all, recall from the proof of Lemma 3 that  $\mathbf{1}^T(\delta + z - z') = 0$ ,  $\delta + z \succeq -1$  and that  $V := \mathbf{1}^T(z + \delta)$  where  $V := |Y|$ . Next, define  $r_i := \delta + z - \sum_{k=1}^{i-1} e_k$ ,  $i = 1, 2, \dots$  and note that update  $e_i \in \arg \min_{e \in E} \|r_i - e\|_\infty$  decreases the largest component of vector  $r_i$ , i.e., in each iteration a component of vector  $r_i$  decreases by 1, and therefore  $\mathbf{1}^T r_i = V - i + 1$  with  $i = 1, \dots, |Y| + 1$ .

For the lower bound observe that if  $r_{i+1}(j) < -1$  for some  $j = 1, \dots, |Y|$  we must have that  $r_i \prec 0$  since the update  $e_i \in \arg \min_{e \in E} \|r_i - e\|_\infty$  selects to decrease the largest component of vector  $r_i$ . However, since  $\mathbf{1}^T r_i \geq 0$  for all  $i = 1, \dots, |Y| + 1$  we have that vector  $r_{|Y|}$  has at least one component that is nonnegative. Therefore,  $r_{|Y|+1} \succeq -1$  and  $\delta + z - z' \succeq -1$ . For the upper bound define  $a_i = -[-r_i]^+$ ,  $b_i = \lfloor r_i - a_i \rfloor$ ,  $c_i = r_i - a_i - b_i$ ,  $i = 1, \dots, |Y| + 1$  and note that  $-1 \preceq a_i \preceq 0$  and  $0 \preceq c_i \prec 1$  for all  $i = 1, \dots, |Y| + 1$ , and that  $\mathbf{1}^T b_i$  decreases by 1 in each iteration if  $\mathbf{1}^T b_i \geq 1$ . Hence,  $b_{|Y|+1} = 0$  and therefore  $-1 \preceq r_{|Y|+1} = a_{|Y|+1} + c_{|Y|+1} = \delta + z - z' \preceq 1$  and we are done.

### C. Proof of Theorem 4

We begin by noting that since  $\mathbf{1}^T u_k = 1 = \mathbf{1}^T e_k$  then  $\mathbf{1}^T s_k = 0$  for all  $k = 1, 2, \dots$ . Also note that since  $u_k \in U$  all elements of  $u_k$  are nonnegative and at least one element must be non-zero since  $\mathbf{1}^T u_k = 1$ .

We now proceed by induction to show that there always exists a choice of  $e_{k+1}$  such that  $s_k \succeq -1$ ,  $k = 1, 2, \dots$ . When  $k = 1$  let element  $u_1(j)$  be positive (as already noted, at least one such element exists). Selecting  $e_1 = v_j$  then it follows that  $-1 < u_1(j) - e_1(j) \leq 0$  and so  $-1 \prec u_1 - e_1 \prec 1$ . That is,  $s_1 \succeq -1$ . Suppose now that  $s_k \succ -1$ . We need to show that  $s_{k+1} \succeq -1$ . Now  $s_{k+1} = s_k + u_{k+1} - e_{k+1}$ . Since  $s_k \succeq -1$ ,  $s_k(j) \geq -1 \forall j = 1, \dots, |Y|$ . Also,  $\mathbf{1}^T s_k = 0$ , so either all elements are 0 or at least one element is positive. If they are all zero then we are done (we are back to the  $k = 1$  case). Otherwise, since all elements of  $u_{k+1}$  are nonnegative then at least one element of  $s_k + u_{k+1}$  is positive. Let element  $s_k(j) + u_{k+1}(j)$  be the largest positive element of  $s_k + u_{k+1}$ . Selecting  $e_{k+1} = v_j$  then it follows that  $s_k(j) + u_{k+1}(j) - e_{k+1}(j) \geq -1$ . That is,  $s_{k+1} \succeq -1$ .

We now show that  $s_k$  is upper bounded. Recall  $e_{k+1}$  can always be selected such that  $s_k \succeq -1$ , and also  $\mathbf{1}^T s_k = 0$ . Since  $\mathbf{1}^T s_k = 0$  either  $s_k$  is zero or at least one element is positive. Since  $s_k \succeq -1$  and at most  $|Y| - 1$  elements are negative, then the sum over the negative elements is lower bounded by  $-(|Y| - 1)$ . Since  $\mathbf{1}^T s_k = 0$  it follows that the sum over the positive elements must be upper bounded by  $|Y| - 1$ . Hence,  $\|s_k\|_\infty \leq (|Y| - 1)$ .

### D. Proof of Corollary 1

Since  $s_k$  has at least one component that is nonnegative, and update (18) selects the largest component of vector  $s_k$  when  $k \in \mathcal{T}$ , we have that a component of vector  $s_k$  can decrease at most by  $\bar{\tau}$  in an interval  $\{\tau_j - \tau_{j+1}\}$  for all  $j = 1, 2, \dots$ . Hence,  $s_k \succeq -\bar{\tau} \mathbf{1}$  for all  $k$ . Next, since  $s_k^T \mathbf{1} = 0$  for all  $k$  and the sum over the negative components is at most  $-\bar{\tau}(|Y| - 1)$ , we have that  $s_k \preceq \bar{\tau}(|Y| - 1)\mathbf{1}$ . The rest of the proof follows as in Theorem 4.

### E. Proof of Theorem 5

Recall that since  $\mathbf{1}^T u_k = 1 = \mathbf{1}^T e_k$  then  $\mathbf{1}^T s_k = 0$  for all  $k = 1, 2, \dots$ , and therefore  $s_k$  is either 0 or at least one of its components is strictly positive. Next, observe that since  $\gamma_k = -\min_{j \in \{1, \dots, |Y|\}} s_k(j)$  we have that  $\max_{j \in \{1, \dots, |Y|\}} s_k(j) \leq \gamma_k(|Y| - 1)$ , which corresponds to the case where  $|Y| - 1$  components of vector of vector  $s_k$  are equal to  $\gamma_k$ . The rest of the proof continues as in the proof of Theorem 4.

## REFERENCES

- [1] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, 1992. [Online]. Available: <http://dx.doi.org/10.1109/9.182479>
- [2] T. Wongpiromsarn, T. Uthacharoenpong, Y. Wang, E. Frazzoli, and D. Wang, "Distributed Traffic Signal Control for Maximum Network Throughput," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*. IEEE, 2012, pp. 588–595.
- [3] S. T. Maguluri, R. Srikant, and L. Ying, "Stochastic models of load balancing and scheduling in cloud computing clusters," in *INFOCOM, 2012 Proceedings IEEE*, March 2012, pp. 702–710.
- [4] M. J. Neely, "Stock market trading via stochastic network optimization," in *49th IEEE Conference on Decision and Control (CDC)*, Dec 2010, pp. 2777–2784.
- [5] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [6] A. Nedić and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1757–1780, 2016/07/09 2009. [Online]. Available: <http://dx.doi.org/10.1137/070708111>
- [7] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American statistical association*, vol. 58, no. 301, pp. 13–30, 1963.
- [8] V. Valls and D. J. Leith, "Dual Subgradient Methods Using Approximate Multipliers," in *Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on*, 2015.
- [9] S. Meyn, *Control techniques for complex networks*. Cambridge University Press, 2008.
- [10] B. Bernstein, "Proof of caratheodory's local theorem and its global application to thermostatics," *Journal of Mathematical Physics*, vol. 1, no. 3, pp. 222–224, 1960.
- [11] M. J. Neely, "Stability and Capacity Regions of Discrete Time Queueing Networks," *ArXiv e-prints*, Mar. 2010.
- [12] A. L. Stolyar, "Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm," *Queueing Systems*, vol. 50, no. 4, pp. 401–457, 2005. [Online]. Available: <http://dx.doi.org/10.1007/s11134-005-1450-0>
- [13] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," in *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, vol. 3. IEEE, 2005, pp. 1794–1803.
- [14] V. Valls and D. J. Leith, "Max-weight revisited: Sequences of nonconvex optimizations solving convex optimizations," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2676–2689, Oct 2016.
- [15] G. D. Çelik and E. Modiano, "Scheduling in networks with time-varying channels and reconfiguration delay," *IEEE/ACM Transactions on Networking*, vol. 23, no. 1, pp. 99–113, Feb 2015.

- [16] M. J. Neely, E. Modiano, and C. P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 2, pp. 396–409, April 2008.
- [17] F. Kelly and E. Yudovina, *Stochastic Networks*. Cambridge University Press, 2014.
- [18] R. Srikant and L. Ying, *Communication networks: an optimization, control, and stochastic networks perspective*. Cambridge University Press, 2013.
- [19] L. Huang, "Deterministic mathematical optimization in stochastic network control," Ph.D. dissertation, University of Southern California, 2011.
- [20] M. J. Neely, "A Simple Convergence Time Analysis of Drift-Plus-Penalty for Stochastic Optimization and Convex Programs," *ArXiv e-prints*, Dec. 2014.
- [21] M. G. Markakis, E. Modiano, and J. N. Tsitsiklis, "Max-weight scheduling in queueing networks with heavy-tailed traffic," *IEEE/ACM Transactions on Networking*, vol. 22, no. 1, pp. 257–270, Feb 2014.
- [22] B. Polyak, "Subgradient methods: A survey of soviet research," in *Nonsmooth optimization: Proceedings of the IIASA workshop March, 1977*, pp. 5–30.
- [23] Y. M. Ermoliev, "Methods of solution of nonlinear extremal problems," *Cybernetics*, vol. 2, no. 4, pp. 1–14, 1966.
- [24] N. Z. Shor, *Minimization methods for non-differentiable functions*. Springer Science & Business Media, 2012, vol. 3.
- [25] A. Nemirovskii and D. Yudin, "Cezare convergence of gradient method approximation of saddle points for convex-concave functions," *Doklady Akademii Nauk SSSR*, pp. 1056–1059, 1978.
- [26] N. Shor, *Minimization methods for nondifferentiable functions*. Translated from Russian by K.C. Kiwiel and A. Ruszczyński, Springer, Berlin, 1985.
- [27] T. Larsson and Z. Liu, "A lagrangean relaxation scheme for structured linear programs with application to multicommodity network flows," *Optimization*, vol. 40, no. 3, pp. 247–284, 1997. [Online]. Available: <http://dx.doi.org/10.1080/02331939708844312>
- [28] D. P. Bertsekas, *Nonlinear Programming: Second Edition*. Athena Scientific, 1999.
- [29] A. Nedić and D. P. Bertsekas, "The Effect of Deterministic Noise in Subgradient Methods," *Mathematical Programming*, vol. 125, no. 1, pp. 75–99, 2010.